AD-A011 230

MODELING THE IMPACT RESPONSE OF BULK CUSHIONING MATERIALS

Don McDaniel

Army Missile Research, Development and Engineering Laboratory
Redstone Arsenal, Alabama

9 May 1975

182159

ADA011230

TECHNICAL REPORT RD-75-16

MODELING THE IMPACT RESPONSE OF BULK CUSHIONING MATERIALS

Don McDaniel
Aeroballistics Directorate
US Army Missile Research, Development and Engineering Laboratory
US Army Missile Command
Redstone Arsenal, Alabama 35809

9 May 1975

U.S. ARMY MISSILE COMMAND

Redstone Arsenal, Alabama

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RD-75-16	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD-A011 230
4. TITLE (and Subtitle) MODELING THE IMPACT RESPONSE OF BULK CUSHIONING MATERIALS		5. TYPE OF REPORT & PERIOD COVERED Technical Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Don McDaniel		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aeroballistics Directorate US Army Missile Res, Dev and Engineering Lab US Army Missile Command Redstone Arsenal, Alabama 35809		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE 9 May 1975 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

PRICES SUBJECT TO CHANGE

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Cushioning Materials
Shock Mitigation
Viscoelastic Properties
Impact Response

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report deals with the utilization of bulk cushioning materials in shock mitigation systems. The current techniques used in designing bulk cushioning systems are discussed, and an improved technique is presented through the development of a mathematical model of impact response which is based on the viscoelastic properties of bulk cushioning materials. A General Model of impact response which is applicable to all types of bulk cushioning materials and predicts g-level response in terms of drop height, static stress,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 Abstract continued

thickness of cushion, and temperature is developed. A technique for determining the optimal cushioning system design is developed, and examples of the use of the technique are presented for a cross-linked polyethylene foam cushioning material.

CONTENTS

Chapter I

INTRODUCTION

A cushioning system is an interface device which isolates a protected item from the shock loads which develop in its external environment. A cushioning system is required when the protected item cannot survive the shock loads imposed by the external environment, unless these loads are attenuated by some form of cushioning system. The design of a cushioning system, therefore, involves stipulations concerning both the item to be protected and the external environment. These two stipulations of how much shock an item can survive and how much shock it is expected to experience in its environment are questions that must be answered in the design of any cushioning system, commercial or military. It is the concern for extreme external environments that is emphasized most in the military designs.

The Military Environment

A cushioning system designed for military use, such as a shock isolating component of a missile or missile system, is required to perform its basic functions throughout its operating life. Current military policy, based on the need for worldwide deployment, dictates that military materiel be capable of withstanding the rigors of deployment on a worldwide basis. Consequently, military materiel, including

cushioning systems, must withstand all the environments to which they are exposed and continue to operate satisfactorily if the materiel is to be considered acceptable.

The environment in which military materiel operates is frequently quite severe and may produce substantial damage. Military operations encompass all the geographic regions and the peculiar aspects of their environments. Each environment introduces hazards peculiar to geographic zones which materiel will encounter when deployed.

The tropics present a hot, humid atmosphere, subjecting materiel to what is commonly referred to as "jungle-rot". Excessive rainfall, high humidity, heat, and fungus, in addition to the prevalence of vermin, insects, and reptiles present problems relating to materiel protection and to the safety of personnel.

The arctic regions subject materiel to a very cold atmosphere. Temperatures of -65°F are common and temperatures of -85°F have been recorded in underground ammunition storage "igloos". The physical characteristics of components, in particular resilient cushioning systems, are radically altered under these conditions and their operational integrity is jeopardized.

Also, high-speed, fast-climbing aircraft introduce rapid and severe variations in temperature and pressure, and provisions must be made to withstand the effects of these changes, when required.

Cushioning Systems in the Military

In evaluating the environmental factors, the military has established certain extreme limits on the particular conditions that have become accepted as a basis for worldwide environments. Temperature

extremes, which produce dramatic changes in the resiliency of cushioning materials, are important in cushion system design. The worldwide temperature extremes are generally defined as -65°F and 160°F; cushioning systems in military applications are expected to perform their shock interface function under these adverse environments.

Many of the military cushioning systems use some form of bulk cushioning material (e.g. plastic foam) as the cushioning system. This provides a low-cost, lightweight cushioning system that is easily incorporated into the design; however, designers have some reservations concerning the use of bulk cushioning systems in the military. One limiting factor is the difficulty the designer has in predicting the response of bulk cushioning materials when subjected to the extreme environmental conditions encountered under military deployment. The temperature sensitivity of these cushioning materials, which causes variations in the impact response, is also a primary concern. It was shown in a recent study of thermoplastic foam cushioning systems that temperature had a significant effect on cushion system response [1]. Therefore, temperature effects must be factored into the design of cushion systems using bulk cushioning materials.

To fully utilize bulk cushioning systems, designers require sufficient information to accurately predict response variations. The current practice is to provide the designer with cushioning data for each type and thickness of cushioning material. These data are provided in the form of dynamic cushioning curves as prescribed in the current theory of cushioning design (Chapter II). For any particular shock isolation system design program, the designer is generally given a maximum allowable fragility level which the protected item is permitted

to experience. Also the particular organization involved will have an established testing policy defining appropriate impact tests. These parameters provide the basis for the design of the shock mitigation system. Given a satisfactory prediction of impact response for various candidate cushioning materials, a designer can use this prediction to select the appropriate cushioning scheme to meet the particular design requirements.

Research Objective

The research objective of this study is to develop a reliable impact response model that accurately predicts the dynamic cushioning performance of bulk cushioning systems. A secondary objective is to develop an optimization technique that utilizes the model in determining an optimal cushioning system design.

In answering the research objective a systematic study of background material was conducted. The current theory of cushioning design is discussed in Chapter II. The ingredients of the General Model of impact response are identified in Chapter III and the basis of the underlying structure of the developed model is based upon viscoelastic theory.

The modeling process and the analysis techniques used to formulate the General Model of impact response are also discussed in Chapter III. Chapter IV presents the two finalized models, a General Model of impact response and the Minicel Model, the model of impact response of a particular cushioning material (Minicel is a 2 lb/ft^3 cross-linked polyethylene foam manufactured by Hercules, Inc).

The validity of the models are demonstrated in Chapter V through a systematic series of tests and analysis. Finally, in answer to the secondary objective of the research, an optimization technique is presented in Chapter VI that generates the optimal bulk cushion design. The optimization technique utilizes the predictive capability of the General Model of impact response to determine optimality and provide, as output, a set of dynamic cushioning curves at the optimal conditions.

Chapter II

CUSHIONING DESIGN THEORY

Anything that is subject to movement is subject to mechanical damage due to shock. Shock may be defined as a sudden change in direction or velocity of the motion of a body. The magnitude or intensity of shock is expressed in G's, which is defined in terms of the time rate of change of the velocity (acceleration), and is measured in feet/second/second. Mathematically, G's = a/g where a is the acceleration experienced by the body and g is the acceleration due to gravity (32.2 ft/sec/sec).

A given body in a static condition has one gravity unit of G acting on it and, therefore, exerts one G upon its support. If this body is raised and allowed to fall freely, it will accelerate in its fall, due to the force of gravity, until it collides with its support or the earth. The stop causes the body to experience a sudden deceleration that can be expressed in terms of G's. If the body experiences, upon impact, a deceleration of 20 times that of its static condition, it is said to have experienced 20 G's of deceleration. A jet pilot experiences such a condition when he pulls out of a dive. He must not exceed his G limit or he will black out, and if the aircraft is not designed and stressed to withstand high G's during such maneuvers, severe damage will occur to the aircraft and it may crash. This same situation exists in regard to fragile objects. The fragility level of an object, measured

in G's, is its ability to withstand deceleration. The purpose of a cushioning system is to reduce the shock transferred to the protected object to a degree below its fragility level, thus protecting it against physical damage.

Cushioning System Design

To understand how a cushioning material functions during shock transfer, one can consider what happens when a body is dropped onto a rigid surface such as a concrete floor. At impact the body is falling at a velocity ($V = \sqrt{2\ gh}$), where V is the velocity at impact in feet/second, g is the acceleration due to gravity of 32.2 feet/second/second, and h is the height of drop measured in feet. In a very short time after impact, this velocity is reduced to zero in a very small distance. Thus, a rapid decrease in velocity occurs, due to impact, and the body is subjected to a very high deceleration.

If the same situation is considered except that the body is dropped onto a resilient cushioning material which rests on the same rigid surface, the body has the same velocity at impact. However, the time required for the velocity to be reduced to zero is much greater than in the previous situation; the rate at which the velocity decreases is considerably less; and the distance traveled afte initial contact with the cushion until the time the velocity is reduced to zero is considerably greater. Compared to the first situation, the body is subjected to lower G's. The resilient cushioning material has, therefore, attenuated the shock pulse by dissipating the kinetic energy present in the body at the time of impact over a longer time period. This has been expressed mathematically [2] as follows:

$$G = \frac{72}{t}\sqrt{h} \qquad \text{(II-1)}$$

where

G = acceleration G - level

t = shock pulse rise time in milliseconds

h = drop height in inches.

Equation (II-1) shows that as the shock pulse rise time is increased, the G's experienced by the body are proportionally decreased.

Increases in shock pulse rise time can usually be accomplished by increasing the thickness of the resilient cushioning material. For example, the G's experienced by a body falling on a tangentially elastic cushioning material can be predicted as follows [2]

$$G = \frac{3.9\ h}{T} \qquad \text{(II-2)}$$

where T = thickness of cushion in inches.

Equation (II-2) shows that the predicted G levels are reduced proportionally with increases in the thickness of the cushioning material. If the cushion thickness is increased, then, during impact, the excursion envelope of the body is increased proportionally and the body moves through an additional amount of cushioning material before it comes to rest. This increase in excursion directly increases the shock pulse rise time with an accompanying reduction in G levels. This reduction is consistent with the change in G levels that would be predicted by Equation (II-1) with an increase in shock pulse rise time.

History of Cushion System Design

Equation (II-2) is based on Mindlin's work in 1945 [3] that marked the beginning of the scientific approach to cushion design. Mendlin's paper was followed by a number of discussions [4-8] by author's who adopted his procedures. The next substantial step forward was the

development of optimum efficiency design points by Janssen [9]. Janssen utilized material properties determined by quasistatic means, in particular, stress-strain curves determined on a conventional compression tester with the speed of compression quite slow (not more than 2 inches/minute). He derived a cushion factor, "J", for several cushioning materials that was the ratio of optimum stress to optimum strain. Then G-level can be predicted on the basis of

$$G = J\frac{h}{T} \qquad \text{(II-3)}$$

where J = Janssen's cushion factor.

This was a significant improvement over Equation (II-2) in that the J values allowed for different performance factors for different materials. However, it soon became apparent that single curves based on the static stress-strain characteristics of a material did not describe the dynamic behavior. Several methods of presentation, all involving families of curves, were attempted. Gradually, Kerstner's approach [10] became the most popular. In this approach, a family of curves is drawn for each material thickness at each drop height, plotting peak dynamic stress (or acceleration) against the original static stress. A typical set of such curves, taken from Humbert and Hanlon [11], is shown in Figure 1.

These curves, referred to as dynamic cushioning curves, are generated for a particular type and thickness of cushion by performing drop tests using standard weight specimens that are dropped onto the cushion.

The static stress (σ_s) is determined by:

$$\sigma_s = \frac{W}{A} \quad , \qquad \text{(II-4)}$$

where σ_s is the static stress (psi), W is the specimen weight, and A is

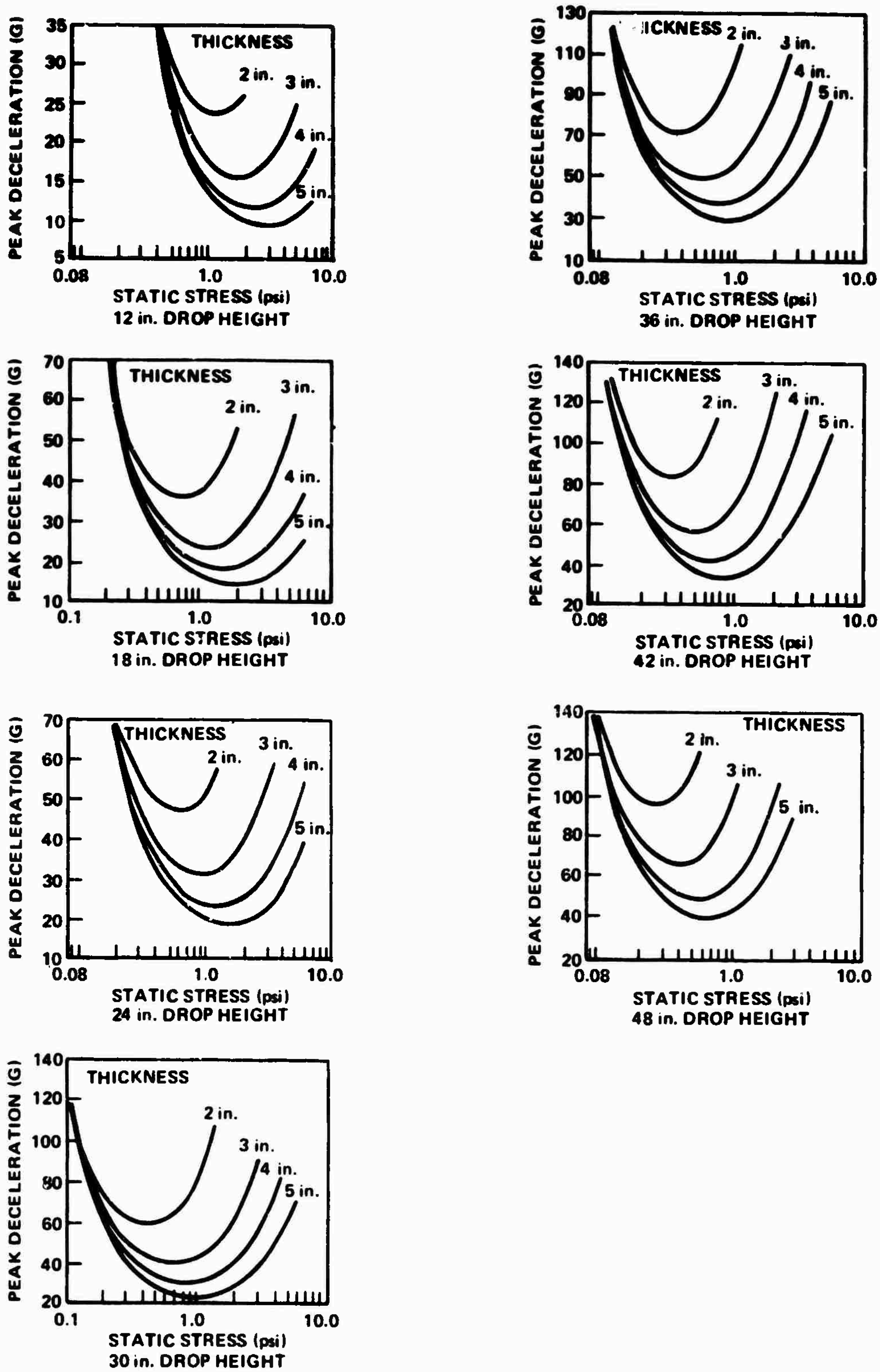


Figure 1. Peak-acceleration versus static-stress curves for polyethylene foam [11].

the footprint of the specimen in the cushion (in.^{2}). A different curve is required for each drop height, thickness, and type of material. The curves are a good indication of the protection to be expected for a particular cushion scheme.

The current practice utilizes the Kerstner procedure in that manufacturers of cushioning products demonstrate the cushioning performance capabilities of their particular products by generating a series of dynamic cushioning curves at various drop heights and thicknesses of cushions (Figure 1). Also, Military Handbook -304, "Cushioning Design Handbook", was published in 1964 which provides families of dynamic cushioning curves for a large number of frequently utilized cushioning materials. Additional work was done by Mustin [12] who used a single value for correlating each dynamic cushioning curve in a family of curves such as Figure 1. This proved to be an over simplification in developing a general model of impact response, as will be seen in Chapter IV.

Reservations Concerning Current Cushioning System Design

In recent years, equipment designers have become increasingly aware of the detrimental effects of extreme temperature upon equipment performance. Consequently, there are now included in the qualification tests of equipment, some tests conducted at temperatures that are representative of the temperature extremes that the equipment is likely to encounter.

This extreme temperature testing has received substantial attention in the military, where the range of temperatures encountered is quite extreme, and the failures can produce catastrophic results. The transportation of military equipment is one area of concern and a study was made of several military containers that used bulk cushioning sys-

tems. It was found that temperature appeared to have a significant effect on impact response [1]. The cushioning systems in the containers did not perform properly due to changes in the performance of the bulk cushioning materials that were induced by extreme temperatures. Consequently, the items packaged in the containers (guided missile systems and system components) did not receive adequate protection, and the missile system reliability was compromised. This type of failure is a potential problem that can occur in very expensive equipment and produce a malfunction in weapon systems that compromises the combat power of a military organization. Also, if proper failsafe provisions are not incorporated into the protection of ordinance items, the safety of any of the personnel that handle the equipment within the logistics system is jeopardized.

Rather than incorporate additional protection into equipment, it is much more cost effective to improve the reliability of cushioning systems. This can be done if a reliable method of predicting cushioning performance can be developed.

A technique has been suggested for modifying the conventional dynamic cushioning curves in such a manner as to address the effect of temperature on the impact response of bulk cushioning systems [1]. The technique utilizes superimposed dynamic cushioning curves that are a super-positioning of the dynamic cushioning curves at temperature extremes upon the ambient dynamic cushioning curve. One such curve is presented in Figure 2. This type of curve demonstrates the effect of temperature for a selected set of conditions and provides the cushioning system designer with the capability of designing cushioning systems with a reduced likelihood of design failure under extreme temperature conditions.

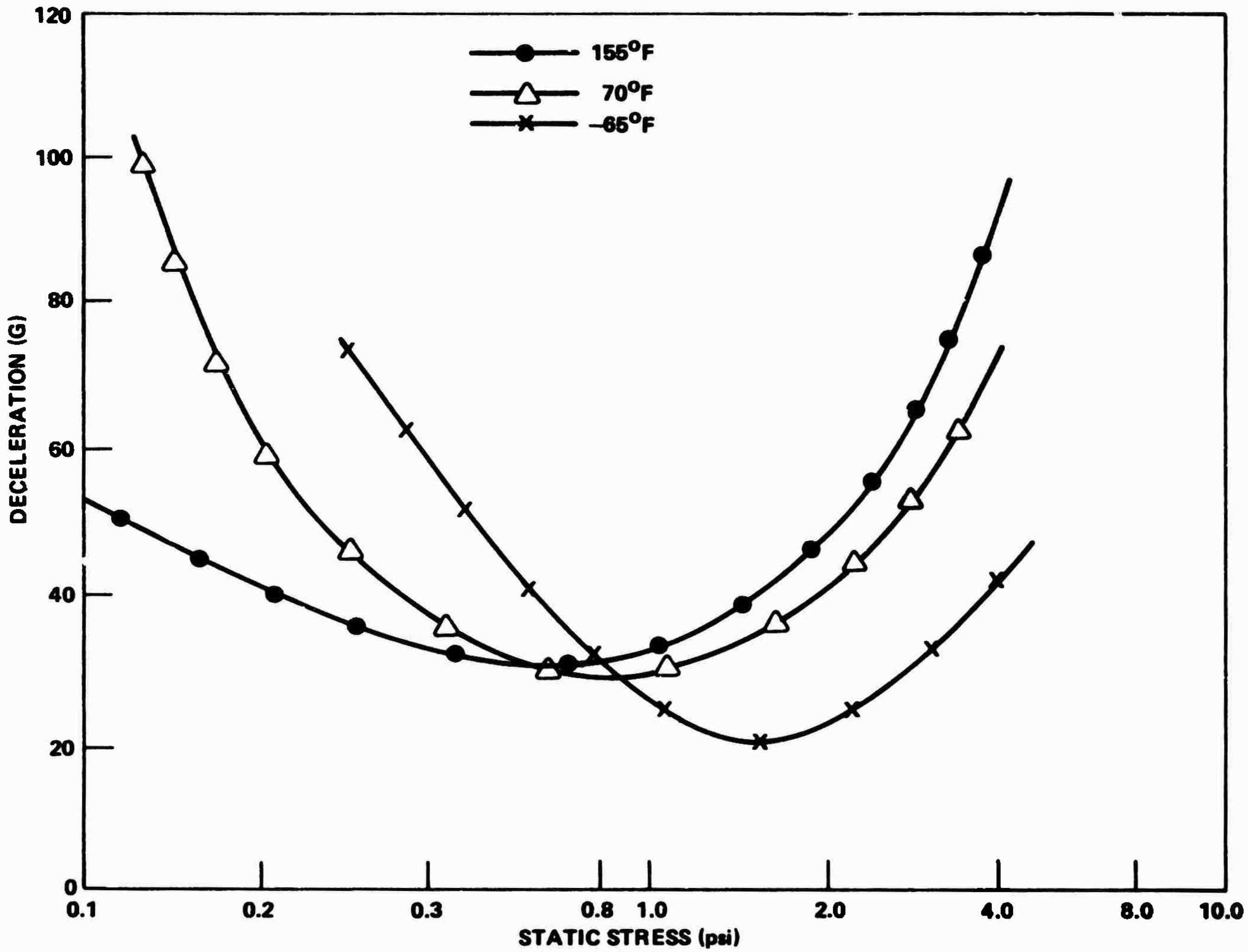


Figure 2. Superimposed dynamic cushioning curve for polyethylene foam (2 lb/ft^3 density, -65°, 70°, and 155°F, 30-in. drop height, 4-in. thickness).

Chapter III

BASIS FOR THE IMPACT RESPONSE MODEL

A valid model of impact response must incorporate all parameters that are expected to have a significant effect on impact response. Temperature has a significant effect on impact response [1], and it is postulated that viscoelastic theory can be utilized to formulate a model of impact response that incorporates temperature effects. The current design practice for predicting impact response is predicated on dynamic cushioning curves. Dynamic cushioning curves do not account for temperature effects on impact response. To improve the predictability of cushioning systems, a model of impact response must account for temperature effects. The temperature effects would be expected to be the most dramatic as the temperature tends towards the extremes; therefore, the temperature extremes encountered by a cushioning system are of particular concern in modeling impact response.

Temperature Extremes in the Model

Army Regulation 70-38, "Research, Development, Test and Evaluation of Materiel for Extreme Climatic Conditions," requires that the extreme external environments that are likely to be encountered, be considered in the design of Army materiel. The temperature extremes to be used in the design are defined in the AR according to the intended deployment of the materiel being designed. Figures 3 and 4 (from AR 70-38) present maps of the extreme temperature conditions to be used in the

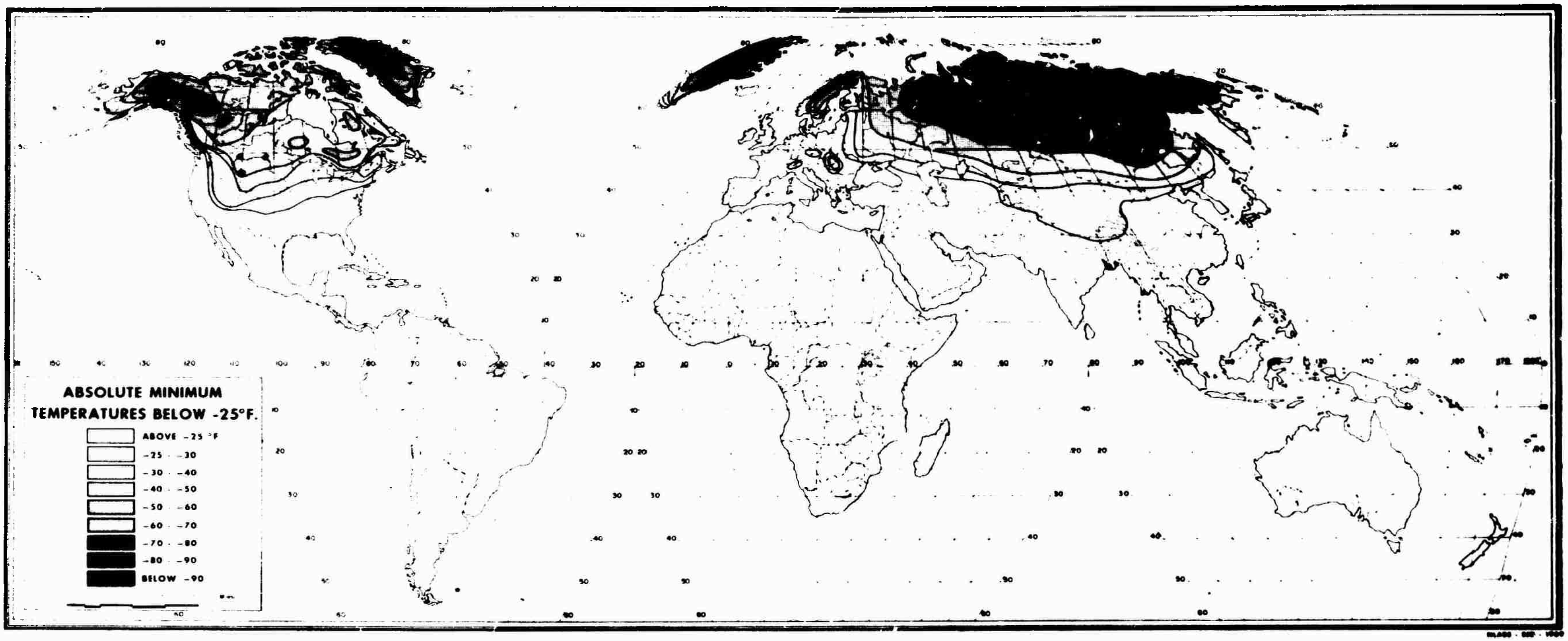


Figure 3. Distribution of absolute minimum temperatures.

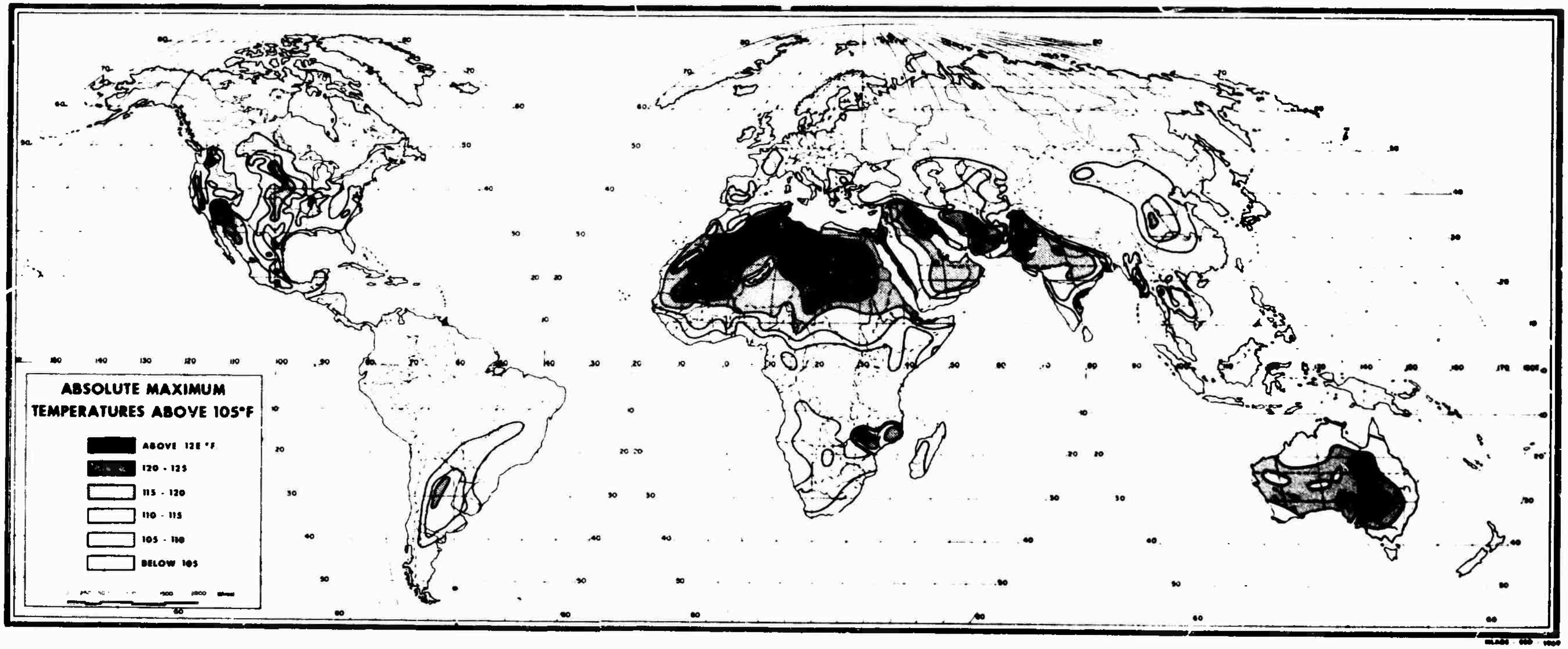


Figure 4. Distribution of absolute maximum temperatures.

design of Army materiel. For example, for worldwide deployment, the operational temperature range is given as -65° to 160°F. However, for continental United States deployment only, the range is from -30° to 145°F. Army materiel is expected to function properly within these ranges and consequently the qualification testing of Army materiel is conducted at these extremes.

A research effort was initiated at the Army Missile Command (MICOM) in 1973 to develop superimposed dynamic cushioning curves that address the effect of temperature on the impact response of cushioning materials. The drop tests were conducted at MICOM and the results analyzed by the University of Alabama in Huntsville (UAH) under a supporting research contract [13]. The initial experimentation was conducted on Minicel material, a cross-linked polyethylene foam material with a 2 pound/foot^3 density manufactured by Hercules, Inc.

The drop test program that was conducted on the Minicel material used drop heights of 12, 18, 24, and 30 inches; temperatures of -65°, 70°, and 160°F; cushion thicknesses of 1, 2, and 3 inches; and static stress levels that varied from 0.04 to 5.0 psi. The G-level response and shock pulse duration were recorded for each of 2736 drop tests. An automated data handling system that included outlier tests and other statistical analyses was developed and used to analyze the Minicel data.

This experimental effort resulted in the generation of a data base of 2409 statistically valid data points. A family of second order polynomial equations was found to be the best predictor of impact response of the Minicel cushioning material. The data are given in Appendix A together with the families of regression polynomials for the 12 inch and 24 inch drop heights, and the correlation coefficients.

Theoretical Basis of the Model

The construction of an impact response model for cushioning materials at varying temperatures requires the development of a functional relationship of the variables. The required relationship can be expressed mathematically as follows:

$$G = F(\sigma_s, T, \theta, h) \tag{III-1}$$

where

G = acceleration G-level

σ_s = static stress in psi

T = thickness of cushion inches

θ = cushion temperature in °F

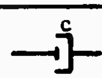
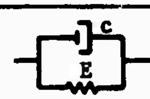
h = drop height in inches.

After considerable research, it was determined that the theory of viscoelasticity could be utilized to provide a theoretical basis for the model. Viscoelastic theory recognizes cushioning materials as belonging to a class of materials which have mechanical properties that are common to perfect solids and perfect liquids. Various theories have been developed over the past century for describing the behavior of perfect solids and perfect liquids. Among these, the oldest theories are the classical theory of elasticity and the theory of hydrodynamics. The classical theory of elasticity deals with the behavior of solids for which the stress is directly proportional to the strain but is independent of rate of strain. This type of solid is known as a Hookian solid (perfect elastic solid). The theory of hydrodynamics describes the behavior of perfect viscous liquids for which, in accordance with Newton's viscosity law, the stress is directly proportional to the rate of strain, rather than the strain itself. In certain instances, solids

and liquids may have their stress related to strain, rate of strain, and higher time aerivatives of strain. Behavior of such materials is termed viscoelastic when the stress is linearly proportional to strain. Materials whose behavior is viscoelastic display solid-like and liquid-like characteristics [14].

Mathematical models have been formulated for viscoelastic materials that have validity over both short and long time periods; for example, Mustin [12] gives the creep and relaxation functions (long term behavior) for a number of simple mathematical models made up of simple spring elements that are assumed linear and massless, and of dashpots in which a piston is moving through a liquid that obeys Newton's law of viscosity (velocity is proportional to strain). The short term stress law and the long term creep and relaxation functions are given in Table I. Creep

TABLE I. SOME CREEP AND RELAXATION FUNCTIONS

Function	Linear	Dashpot	Voigt Solid	Maxwell Solid
Pictorial representation				
Stress law	$\sigma = E\epsilon$	$\sigma = c\,\dfrac{d\epsilon}{dt}$	$\sigma = E\epsilon + c\,\dfrac{d\epsilon}{dt}$	$\dfrac{\sigma}{c} + \dfrac{1}{E}\dfrac{d\sigma}{dt} = \dfrac{d\epsilon}{dt}$
Creep function	$\dfrac{1}{E} \cdot H(t)$	$\dfrac{t}{c} \cdot H(t)$	$\dfrac{1}{E}\left(1 - e^{-Et/c}\right) \cdot H(t)$	$\left(\dfrac{1}{E} + \dfrac{t}{c}\right) \cdot H(t)$
Relaxation function, $R(t)$	$E \cdot H(t)$	$c \cdot \delta(t)$	$E \cdot H(t) + c\,\delta(t)$	$(Ee^{-Et/c}) \cdot H(t)$

Notes:
E = spring constant
σ = stress
ϵ = strain
c = damping coefficient
$H(t)$ = unit step function
$\delta(t)$ = Dirac delta function

functions are shown in Figure 5 while Figure 6 illustrates the relaxation behavior. The appearance of the relaxation functions indicates that the stress is infinite at the instant that strain occurs. This is due to the dashpot element which, unlike a spring, cannot give a finite instantaneous strain response to a finite instantaneous force change. Therefore, an infinite force is required to produce a finite instantaneous strain.

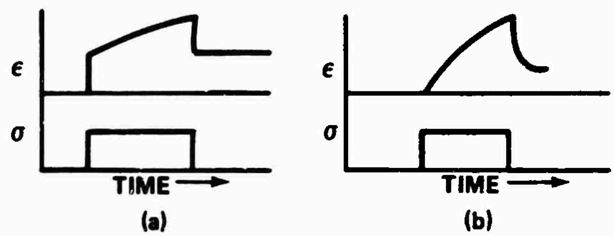


Figure 5. Behavior of some simple creep functions: (a) Maxwell solid, and (b) Voigt solid.

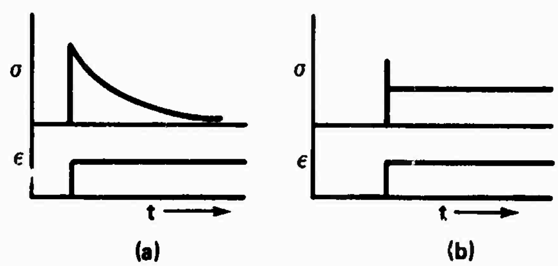


Figure 6. Behavior of some simple relaxation functions (R(t)): (a) Maxwell solid, and (b) Voigt solid.

These relationships are expressed compositely in constitutive equations which, in viscoelastic theory, describe the response of materials to mechanical excitation. The constitutive equation in conjunction with the energy, momentum, and continuity equations permits the prediction of complicated responses to complicated excitations, like the flow of liquids in irregularly shaped channels or the deflections of beams of varying cross sections under complicated loading conditions.

There are two approaches to obtaining the constitutive equation of a material: it can be obtained either experimentally with the help of some simple, well-defined tests (like the stress-strain curve) or the response of the material to some strain or stress history can be calculated with the help of a model describing its structure. Hooke's law is an example for the first group, the phenomenological equations. It is based solely on experimental observation. The best known example in the latter category of structural theories is the kinetic theory of rubber elasticity predicting the elastic response of vulcanized elastomers from their structure.

While the phenomenological constitutive equations describe the results of experimentally obtained data and are usually applied to predict more complicated behavior, the structural ones offer an insight into structure-property relations [15].

Since temperature effects are required in the constitutive equation that models impact response, one must consider that portion of viscoelastic theory which accounts for the behavior of materials at varying temperatures. Most of this theory arises from the consideration of the behavior of the molecule. The molecule may be visualized as a long curled elastic chain which may have cross-linking with other molecular chains. Since molecular activity is a function of temperature, strain response (the summation of individual molecule responses) is also a function of temperature. On this basis, in the glassy zone (the zone below the temperature at which the polymer structure starts to become brittle), molecular "curling and uncurling" cannot occur rapidly enough to follow the stress. In the rubbery zone (the zone

above the temperature at which the polymer structure becomes rubbery), curling and uncurling are in phase with the stress which is not conducive to energy dissipation.

Let the long term elasticity of a material at a given temperature, θ_1 , be plotted against the natural logarithm of time. Suppose, now, that the curve shifts horizontally to the right as the temperature is lowered. This situation is illustrated in Figure 7 for three temperatures.

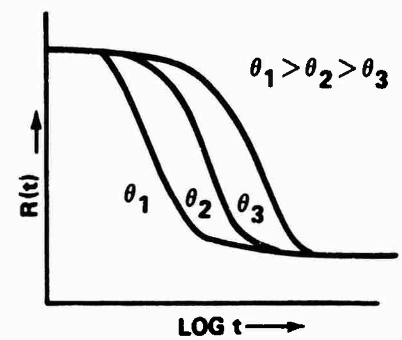


Figure 7. Shift in elasticity characteristics as a function of time and temperature.

The behavior sketched in Figure 7 is typical of many materials utilized as cushions. Materials which behave in this manner are called thermorheologically simple. Amorphous polymers behave in this manner while polymers with crystalline structures do not. Most cushioning materials may be considered amorphous. Teflon and plasticized polyvinyl chloride are not amorphous, but they are rarely used as cushions [16].

Viscoelastic Considerations

A constitutive equation that relates the maximum G-level experienced by a body when subjected to a deceleration impact into a cushioning material is required for the impact response model stated in Equation (III-1). A partial basis is found in a paper on the viscoelastic

properties of thermorheologically simple cushioning materials by Cost [17]. Cost considers a viscoelastic Kelvin body which is identical to the Voight model shown in Table I. The stress laws are identical with Figure 7 except for notation, in that Cost uses K and η for E and c, respectively.

Consider a body of mass M falling under the influence of gravity from a height h and impacting a cushion material of area A_c and thickness T. The static stress σ_s is defined as the weight of the body divided by the area of the cushion A_c.

Assume the cushion material behaves as a viscoelastic Kelvin body whose stress strain relation can be expressed as

$$\sigma = K\epsilon + \eta\dot{\epsilon} \qquad \text{(III-2)}$$

where η and K are material properties. When Equation (III-2) is applied to the problem under consideration, σ becomes the static stress in a bulk cushion, σ_s, and ϵ, the unit strain, can be expressed in terms of the displacement of the upper surface of the cushion as $\epsilon = x/T$. If the principles of Newtonian mechanics are applied to the falling body after time of impact where force equals mass times acceleration, then the equation of motion for the body can be expressed as

$$M\ddot{x} = -\sigma_s A_c \qquad \text{(III-3)}$$

Upon substitution of Equation (III-2) into Equation (III-3)

$$M\ddot{x} = -\frac{A_c K}{T} x - \frac{A_c \eta}{T} \dot{x} \qquad \text{(III-4)}$$

where x is measured relative to the initial location of the upper surface of the cushion and is considered positive downward. Rearranging the equation of motion gives

$$\ddot{x} + \frac{A_c\eta}{MT}\,\dot{x} + \frac{A_c K}{MT}\,x = 0 \quad . \tag{II -5}$$

Equation (III-5) can be cast in the following canonical form:

$$\ddot{x} + 2\beta_c\,\omega_n\,\dot{x} + \omega_n^2\,x = 0 \tag{III-6}$$

provided β_c and ω_n are defined as follows:

$$\beta_c = \frac{\eta g}{2\sigma_s\,T\,\omega_n} = \frac{\eta A_c}{2MT\,\omega_n}$$

$$\omega_n = \sqrt{\frac{gK}{\sigma_s\,T}} = \sqrt{\frac{A_c K}{MT}} \quad . \tag{III-7}$$

The parameters β_c and ω_n have physical significance as being the damping factor and natural frequency of the undamped system, respectively. For the case of "underdamped" motion, the solution of Equation (III-6) can be expressed as

$$x(t) = e^{-\beta_c\,\omega_n\,t}\,[B_1 \cos \mu t + B_2 \sin \mu t] \tag{III-8}$$

where

$$\mu = \omega_n \sqrt{1 - \beta_c^2} \tag{III-9}$$

and B_1 and B_2 are arbitrary constants which can be determined from the initial conditions. Applying the initial conditions

$$x(0) = 0 \quad ,$$

$$\dot{x}(0) = \sqrt{2gh} \tag{III-10}$$

to evaluate the two constants B_1 and B_2 gives

$$B_1 = 0 \quad ,$$

$$B_2 = \frac{\sqrt{2gh}}{\omega_n\sqrt{1 - \beta_c^2}} \quad . \tag{III-11}$$

Therefore, the acceleration history can be expressed as

$$\ddot{x}(t) = \frac{\omega_n \sqrt{2gh}}{\sqrt{1 - \beta_c^2}} \, e^{-\beta_c \omega_n t} \left[(2\beta_c^2 - 1) \sin \omega t - 2\beta_c \sqrt{1 - \beta_c^2} \cos \omega t \right] . \tag{III-12}$$

In this research, only the maximum value of the acceleration is of interest. Consequently, it is necessary to separate the solution into oscillatory and decaying components. To accomplish this, a trigonometric identity must be introduced into Equation (III-12) to obtain the following expression:

$$\ddot{x}(t) = -\frac{\omega_n \sqrt{2gh}}{\sqrt{1 - \beta_c^2}} \, e^{-\beta_c \omega_n t} \cos \left[\omega_n \sqrt{1 - \beta_c^2} \, t + \gamma \right] \tag{III-13}$$

where

$$\gamma = \tan^{-1} \left[\frac{2\beta_c^2 - 1}{2\beta_c \sqrt{1 - \beta_c^2}} \right] . \tag{III-14}$$

Equation (III-13) expresses the acceleration of the falling mass as a function of time. It is observed that the character of the response is that of a damped sinusoid with the exponential and trigonometric components of the solution governing the transient behavior. Since the magnitude of the exponential or the trigonometric term can achieve a maximum value of one, the coefficient of these terms governs the absolute magnitude of the acceleration. Thus, in regard to the form of the equation, it can be seen that

$$\ddot{x}_{max} \approx \frac{\omega_n \sqrt{2gh}}{\sqrt{1 - \beta_c^2}} . \tag{III-15}$$

Recalling the expansion

$$(1 - Z^2)^{-1/2} = 1 + \frac{1}{2} Z^2 + \frac{3}{8} Z^4 + \ldots \quad , \tag{III-16}$$

the peak acceleration can be expressed as

$$\ddot{x}_{max} \approx \omega_n \sqrt{2gh} \left(1 + \frac{1}{2} \beta_c^2\right) \tag{III-17}$$

where only terms up through the second order in β_c have been retained. Returning to the primary variables σ_s, T, h, and θ, and Equation (III-7) gives

$$\ddot{x}_{max} \approx g \left(\frac{hK}{\sigma_s T}\right)^{1/2} \left[1 + \frac{1}{8} \frac{\eta^2 g}{\sigma_s T K}\right] \quad . \tag{III-18}$$

Inserting arbitrary constants instead of the specific numerical coefficients gives

$$\ddot{x}_{max} \approx C_1 \frac{h^{1/2}}{\sigma_s^{1/2} T^{1/2}} + C_2 \frac{h^{1/2}}{\sigma_s^{3/2} T^{3/2}} \eta^2 \tag{III-19}$$

where η is a material property dependent on temperature.

Assuming thermorheologically simple behavior for the viscoelastic material, η can be expressed as

$$\eta(\theta) = \eta_0 \, a(\theta) \tag{III-20}$$

where $a(\theta)$ is the "shift factor". Furthermore, $a(\theta)$ has been shown by experience [16] to be expressible as

$$a(\theta) = C_1' \, e^{-C_2' (\theta - \theta_0)} = C_1'' \, e^{-C_2' \theta} \quad . \tag{III-21}$$

Substituting into Equation (III-20) gives

$$\eta(\theta) = \eta_0 \, C_1'' \, e^{-C_2' \theta} \tag{III-22}$$

and

$$\eta^2(\theta) = C_3 \, e^{-C_4 \theta} \tag{III-23}$$

which can be expanded in a Taylor series to give

$$\eta^2(\theta) = C_3\left[1 - C_4\theta + C_5\theta^2\right] \tag{III-24}$$

where only terms up through the second order have been retained. Upon substitution of this expression into Equation (III-19), we get

$$\ddot{x}_{max} \approx K_1 \frac{h^{1/2}}{\sigma_s^{1/2}\,T^{1/2}} + K_2 \frac{h^{1/2}}{\sigma_s^{3/2}\,T^{3/2}} + K_3 \frac{h^{1/2}}{\sigma_s^{3/2}\,T^{3/2}}\,\theta + K_4 \frac{h^{1/2}}{\sigma_s^{3/2}\,T^{3/2}}\,\theta^2 \tag{III-25}$$

as an expression relating the variables σ_s, T, h, and θ to the peak acceleration.

If additional terms are retained, a more general expression will result. Such a general expression is

$$\ddot{x}_{max} \approx C_0\left(\frac{h}{\sigma_s\,T}\right)^{1/2} + \sum_{i=1}^{N} \frac{C_i h^{1/2}}{(\sigma_s\,T)^{i+1/2}}\left[\sum_{j=1}^{M} K_{ij}\;\theta^j\right] \tag{III-26}$$

where the C_i ($i = 0, 1, 2, \ldots, N$) and K_{ij} ($j = 1, 2, \ldots, M$) are constants to be determined by curve fitting procedures.

Expressions of the form given in Equations (III-25) and (III-26) appear likely candidates for empirically curve-fitting cushioning system experimental data. This concludes Cost's derivation.

Verification

Once a candidate relationship was selected, it was verified. A statistical comparison of the G-level predicted by the function to the data base listed in Appendix A was used to verify the model.

This was accomplished by modifying the Stepwise Regression Procedure given by Draper and Smith [18]. In the Draper and Smith version,

variables are entered and removed from the regression at each stage on the basis of F-tests performed on the coefficients of the independent variables. This procedure was modified in this research in that the variables were entered into the regression equation and remain there as long as they make a contribution to the overall correlation. This modification was considered most appropriate during the preliminary phases of the modeling. It precludes the removal of a variable from the regression equation at one stage because other variables can explain most of the variation, and then find in a later stage that the variable was needed but was not available. In the Draper and Smith version, the establishment of critical F values can inadvertently cause the removal of a desirable variable.

It is considered most important in the early stages of model development that all variables that can possibly be retained remain in solution. Once they are discarded as irrelevant, it is difficult to reincorporate them and there is a substantial risk that they will be lost. However, the modification can cause the stages in the solution to carry superfluous variables through several iterations and ultimately retain them in the final solution. This does require some small amount of additional computer time. However, these unnecessary variables are easily identified and can be discarded during the fine tuning of the final solution with little risk to the validity of the model. If variables are discarded prematurely, which could happen in the Draper and Smith version of the regression analysis, the validity of the model may be compromised.

A program listing of the Stepwise Regression Procedure that was utilized, called MLRD, is given in Appendix B. One form of output from the

program is printer plots in the form of dynamic cushioning curves. These are a nesting of various thicknesses of a particular cushioning at a particular drop height and temperature. The curves are plotted using numbers corresponding to the thickness of cushion. The dynamic cushioning curves that were selected from the UAH study are displayed in Appendix C using the MLRD format. The initial validation exercise for the modeling effort, then, consisted primarily of displaying the curve shape and nesting of the model being validated to determine if the model generated dynamic cushioning curves that compared favorably with those in the UAH study.

Theoretical Validity

The required form of the model as given in Equation (III-1) can be identified in viscoelastic theory as a phenomenological constitutive equation. The model proposed by Cost was selected as the most viable candidate, where $\ddot{x}_{max}$ is the maximum G-level experienced by a body during impact into a cushion, and C_0, C_1, K_1, K_2, etc. are regression constants. This equation was used as the basic underlying structure of the model because it provides a functional relationship of all the variables required in Equation (III-1) and has a theoretical basis in viscoelastic theory. There was one transformation of variables made prior to using the equation. It can be reasoned that when θ is a significant aspect of the model, then for θ^j, where an exponent j is added, the polarity of θ^j will oscillate. To avoid this difficulty, θ was transformed to °R, so the extreme temperatures encountered in the data are always positive.

Validity of Initial Trials

A program code of Equation (III-26) was run on MLRD and the MLRD plot (Figure 8) shows that $\ddot{x}$ was a progressively decreasing value as σ_s increased. This was anticipated to an extent by Cost during his finite element analysis [17]. He generated a plot of peak acceleration versus static stress in the region where stress is proportional to strain (Figure 9).

This deficiency in a constitutive equation is not at all uncommon as Meinecke [15] suggests that even though the behavior of actual materials is usually different from that predicted by various classical theories, for engineering purposes it may be worthwhile to approximate the actual behavior to the idealized behavior. But for cases where it is not possible to approximate the actual behavior of the material to the idealized behavior without sacrificing the accuracy in prediction, it is very essential to consider the anomalies from the idealized character. The departure of actual material from its idealized character may be due to various reasons. For instance, the deviation from the idealized character may be such that the stress, instead of being directly proportional to the strain, is related to it in a complicated manner. Solids display this behavior when they are stressed beyond the so called elastic range or when the deformation becomes so large as to introduce nonlinearity between stress and displacements. Likewise, liquids for which the stress is nonlinearly related to rate of strain are termed non-Newtonian liquids. Equation (III-26) is a good representation of the linearity of impact response but a nonlinearity must be introduced as an initial correction for shape to get a U shaped curve.

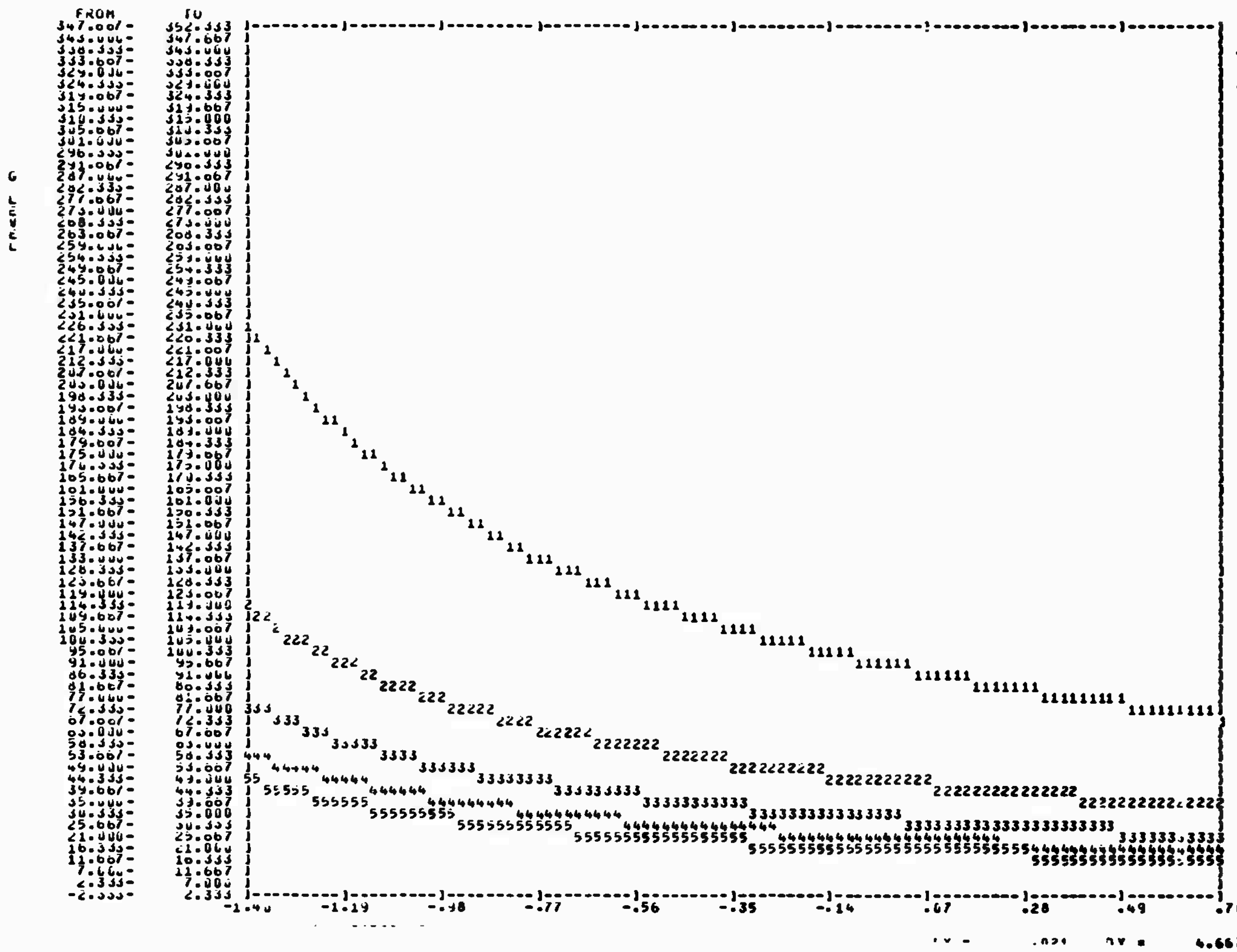


Figure 8. Preliminary MLRD plots of dynamic cushioning curves of the response model at 70°F and 12-inch drop height.

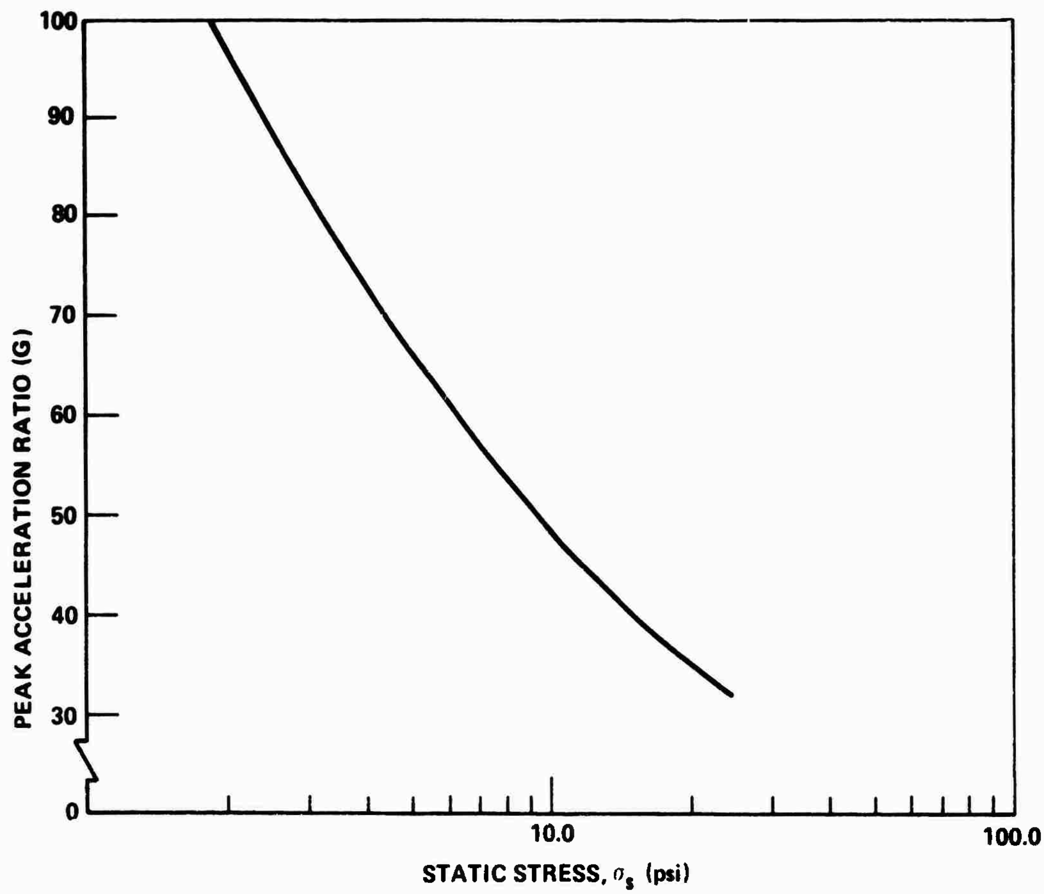


Figure 9. Finite element solution results for peak acceleration.

Initial Correction for Shape

All known dynamic cushioning curves exhibit a characteristic U shape. Consider a compression spring approximately 6 inches in diameter and a free height of 6 inches, similar to those used in the front suspension of an automobile. On this spring a rigid steel plate that serves as an impacting surface is placed. Then, a safety pin, an automobile, and a locomotive are dropped, in turn, onto the spring from approximately 6 inches.

Now the results of the drop test are examined. The compression spring is much too stiff for the small weight of the safety pin, so the

spring does not deflect to store the energy of the fall. This rapid deceleration of the pin causes it to experience high G-levels. Because the compression spring was taken from an automobile, the spring deflects with the fall of the automobile. Energy is stored by the spring and the load on the automobile is proportional to the stiffness of the spring and its deflection. For the locomotive, due to the extremely heavy mass, the spring deflects until the coils of the springs have compressed completely. Up to this point the deceleration has progressed slowly with no high G-levels. However, from this point the locomotive will be decelerated to zero velocity almost immediately and will experience high G-levels.

Thus, for a given stiffness and free height of the spring, the appropriate weight dropped from a given height will result in the maximum energy storage in the spring and a minimum G-level experienced by the weight. The U shapes of dynamic cushioning curves, such as those in Figure 1, are the result of this property of bottoming of bulk cushions. The optimal conditions exist at the ogive of the curve as indicated in Figure 10. At this point the maximum energy is stored in the cushion with the accompanying minimum G level.

A simplified model of a foam material such as the one proposed by Gent and Thomas [19] (Figure 11) is helpful in visualizing the transitional aspects of why bottoming occurs. The foam consist of thin threads joined together to form a cubical lattice. Mechlin [20] explains the bottoming effect as the result of the ligaments of the cushioning material structure packing together, one against another.

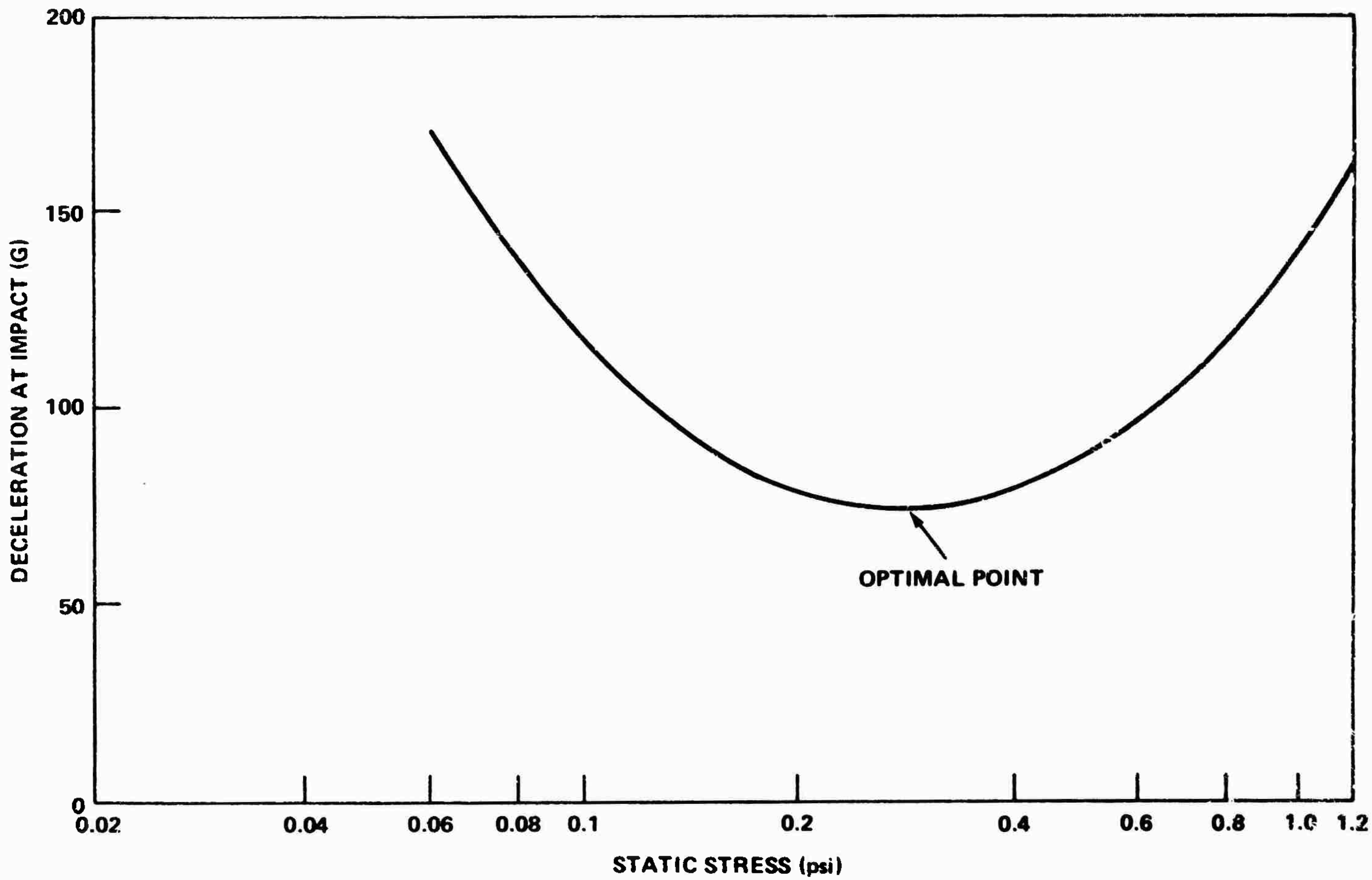


Figure 10. Typical dynamic cushioning curve (Minicel, 30-in. drop height, 70°F, 1 in. thickness).

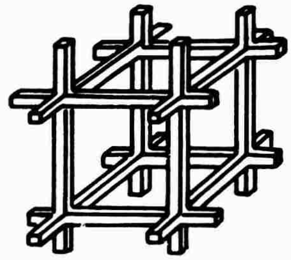


Figure 11. Structural model of a foam material [22].

This situation could be characterized as representing an entirely new material with stress-strain properties substantially different from the original cushioning material. Accordingly, the initial model, Equation (III-26), does not address the non-linearity introduced beyond the bottomed region of the curve, and in itself is not a satisfactory model of cushion response. However, it is reasonable to assume that the functional arrangement of the variables may not be rearranged through the bottomed region of the curve. This possibility was explored using a modular modeling technique and it was found that the basic arrangement of the variables has merit.

Chapter IV

THE IMPACT RESPONSE MODELS

Development of the General Model

The initial model, Equation (III-26), proved deficient in representing the nonlinear characteristics of cushion response. However an extensive literature search showed it to be the only known model that provides a direct relationship of the required variables. Consequently, a modular technique suggested by Shannon [21] was used to modify this model and construct a valid model of impact response. The relationship of each independent variable (σ_s, T, θ, h) and its effect upon the dependent variable (G-level) was studied and the finalized relationship for each independent variable was then incorporated into the model.

Variable 1, Drop Height (h)

The effect of drop height upon G-level that is given in Equation (II-1) is based upon the relationship,

$$V = \sqrt{2gh} \quad . \tag{IV-1}$$

V is the velocity at impact and is related as the square root of drop height (h). Mindlin [3], Janssen [9], and others utilized this relationship of G-level versus drop height, and the derivation by Cost in Equation (III-26) is on this same basis. It was determined that drop height should be incorporated into the model as $h^{1/2}$.

Variable 2, Static Stress (σ_s)

In the UAH study [12], many relationships of G's versus static stress were investigated and it was found that the best agreement was obtained in a second order polynomial of the natural log of stress. Several polynomials of various orders of static stress were tested in this research and a similar conclusion was reached, namely, that a second order polynomial was superior and that the desired U shaped dynamic cushioning curves of G's versus static stress would result.

The initial MLRD Computer runs were made using the following relationship:

$$G = F(\sigma_s, h) \quad . \tag{IV-2}$$

The variables were input with $h^{1/2}$ and a second order polynomial of σ_s. The best fit was obtained using the following functional relationship:

$$G = C_0 + C_1 h^{1/2} + C_2 h^{1/2} \ln \sigma_s + C_3 h^{1/2} (\ln \sigma_s)^2 \quad . \tag{IV-3}$$

These polynomials generated U shaped curves similar to those in the UAH study (compare Figure C-2 and Figure 12).

Variable 3, Thickness of Cushion (T)

Janssen [9], Mindlin [3], and others suggested that G-level was an inverse relationship to thickness as given in Equation (II-2). This seems intuitively correct and many attempts were made using this relationship. T was introduced into these as a negative exponential such as $T^{-1/2}$, $T^{-3/2}$, $T^{-5/2}$, $T^{-7/2}$, etc. Computer runs showed good correlation with thickness input as a negative exponential of the type given. Figure 12 shows the typical nesting effect obtained.

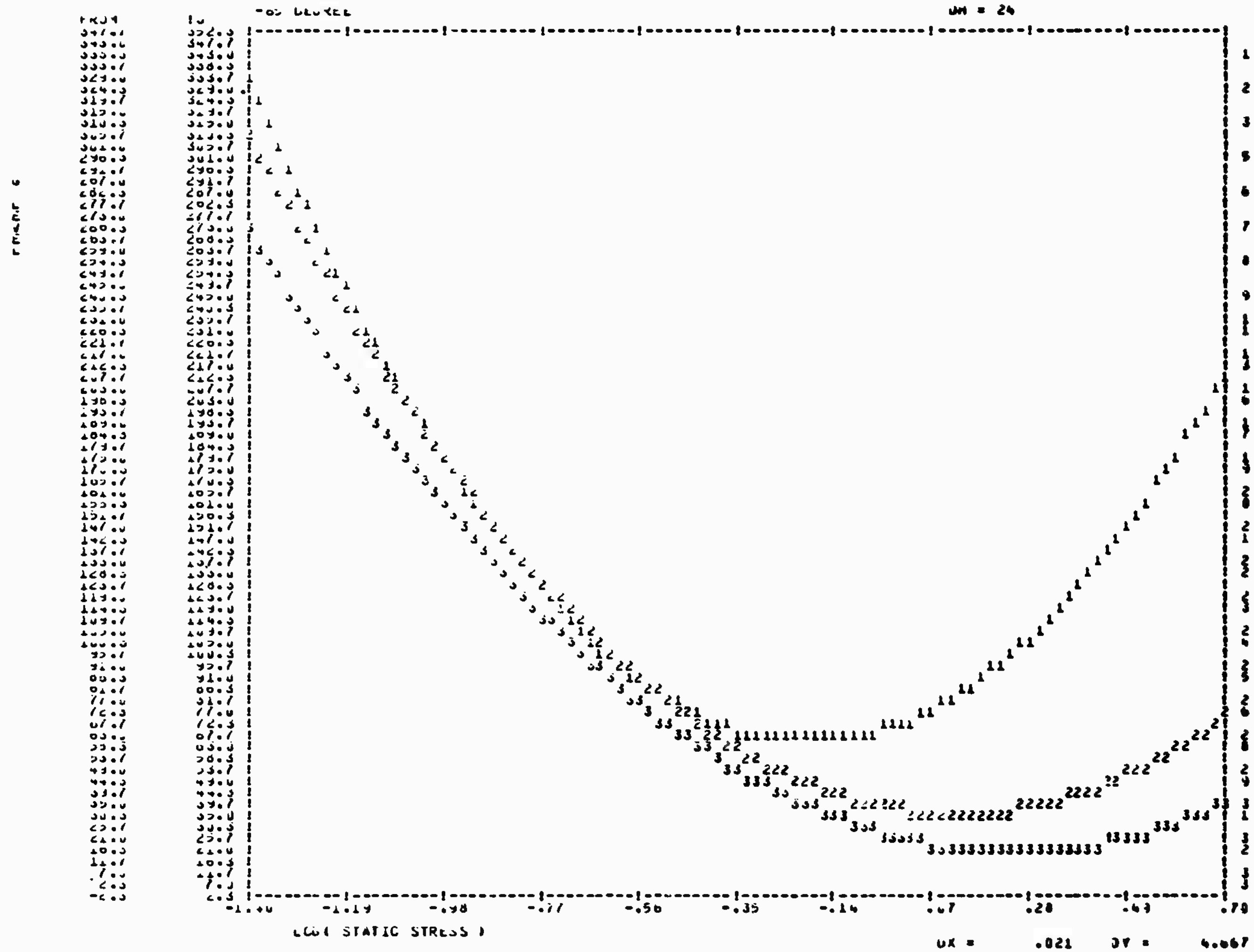


Figure 12. MLRD plots of dynamic cushioning curves of $G = C_0 + C_1\, h^{1/2}\, \ell n\, \sigma_s + C_2\, h^{1/2}\, (\ell n\, \sigma_s)$ at -65°F and 160°F and 24-inch drop height.

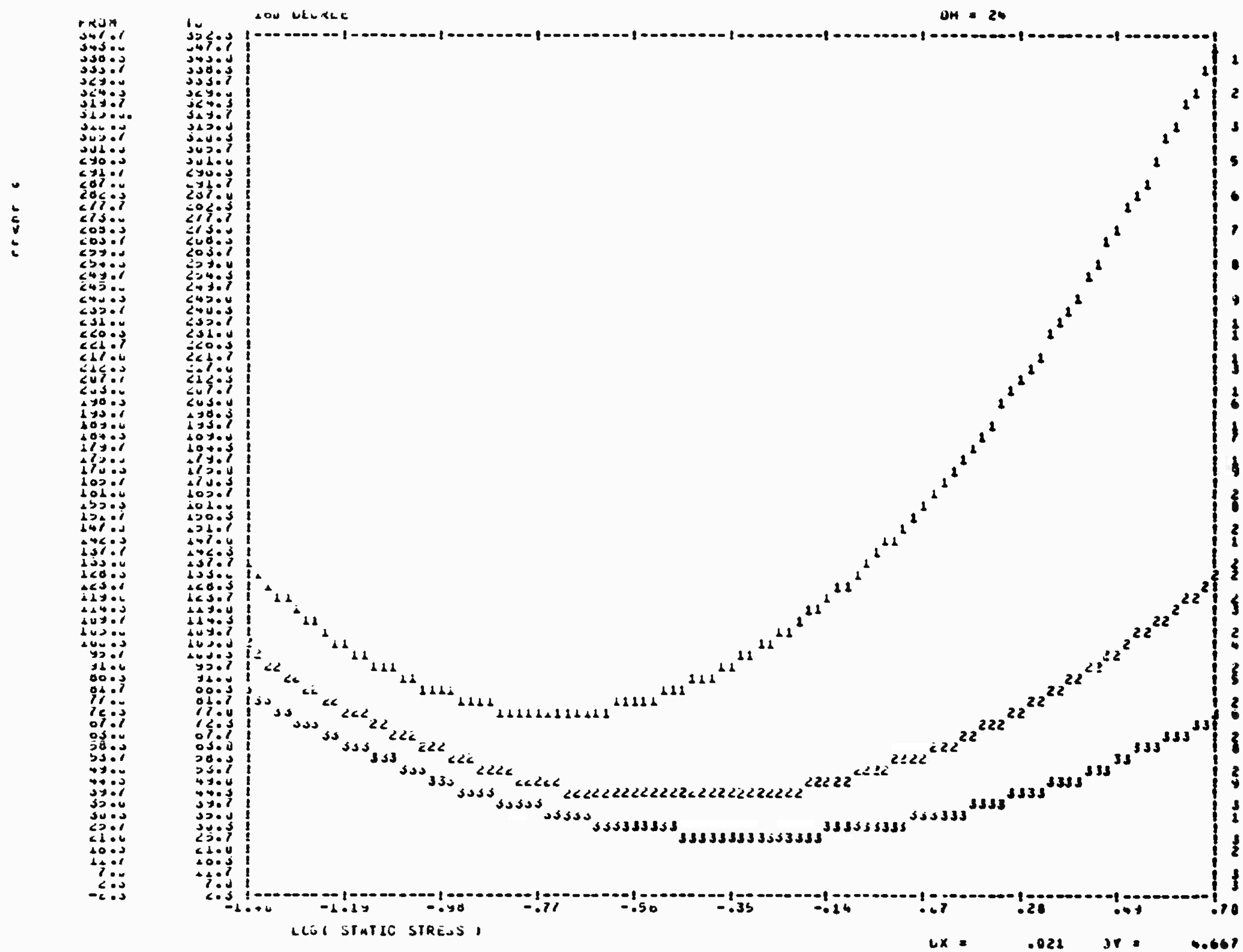


Figure 12. Concluded.

Variable 4. Temperature (θ)

Temperature effects were expected to be the most difficult to incorporate and this turned out to be the case. It can be reasoned that the phase shift effect of temperature discussed in viscoelastic theory for thermorheologically simple materials produces the multiplier effect of the exponentials shown by Cost. Models were tried with several orders of temperature and the polynomial with θ^j, where $j = 1, 2, 3, \ldots, n$ were the most satisfactory. Figure 13 is a typical plot using θ^j (where $j = 1, 2, 3$) that shows the shifting obtained from a cold temperature of -65°F (C) through ambient (A) to hot 160°F (H).

The General Model

Many relationships were tried and rejected. Each time, the basic underlying structure of the variables was rejustified and new formulations were attempted. The process was repeated until a valid General Model of impact response was developed. The General Model is given as follows:

$$G = C_0 + \sum_{\ell=0}^{S} h^{\ell/2} \sum_{k=0}^{R} \frac{1}{T^{(1/2+K)}} \sum_{j=1}^{N} \theta^j \sum_{i=0}^{M} C_{ijk\ell} (\ell n\ \sigma_s)^i \quad . \tag{IV-4}$$

This General Model incorporates each of the variables in the manner prescribed in the modular modeling effort. The curves produced by this model are all U shaped and can be displayed using the MLRD plot routine. A series of plots from the General Model for various temperatures and drop heights are given in Appendix D for the 18 and 30 inch drop heights at -65°, 70°, and 160°F. A comparison of these curves

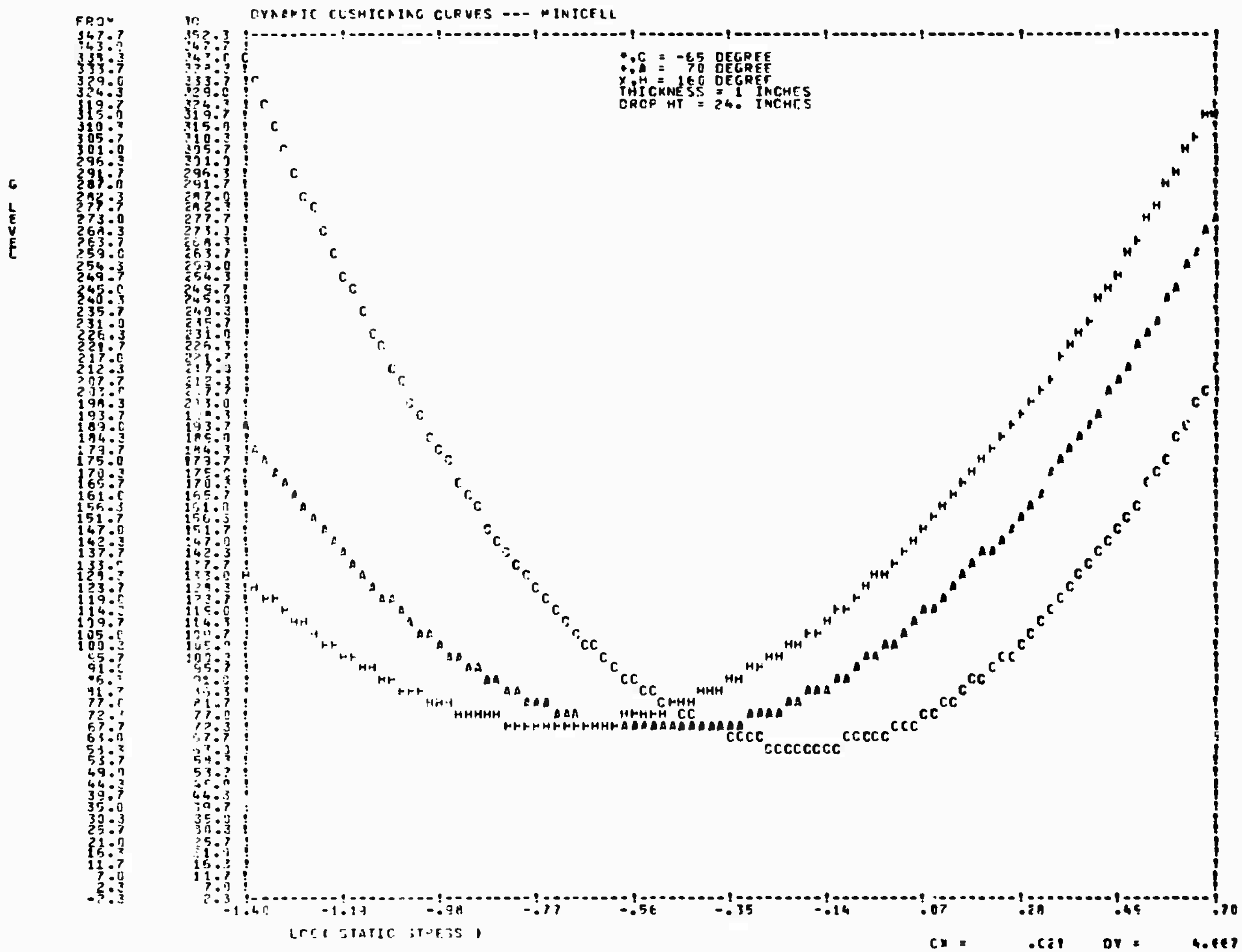


Figure 13. MLRD plots of superimposed dynamic cushioning curves derived from a model using θ^{j}.

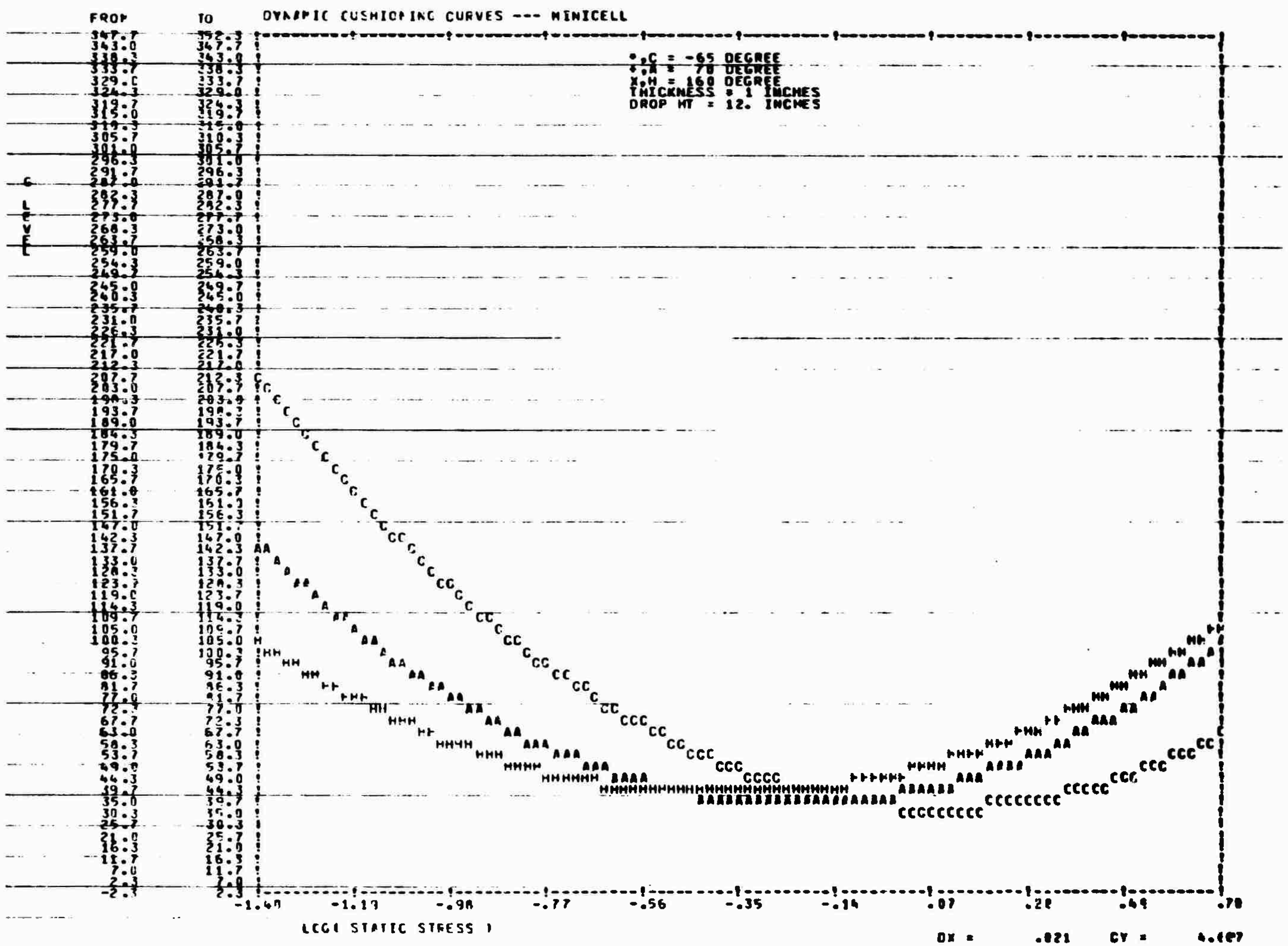


Figure 13. Concluded.

demonstrates the following:

A) The curves of G-level versus static stress are U shaped, consistent with cushioning theory

B) G-level decreases as drop height decreases, consistent with Equation (II-1)

C) The curves are shifted laterally with temperature, consistent with the thermorehologically simple assumption

D) G-level decreases as thickness increases, consistent with Equation (II-2)

E) Thickness curves are nested similar to Humbert and Handlon [11], Figure 1.

Precision of the General Model

The precision of this General Model, Equation (IV-4), has to do with how well it can be made to represent a particular type of cushioning material. This can be determined through comparison with an experimental data base similar to that in Appendix A. Sensitivity analysis can be performed on the model by adjusting the upper limits of the summations, M, N, R, and S to determine if it is possible to obtain the desired precision.

A sensitivity analysis was performed using the Minicel data of Appendix A. S is set at 1, R is set at 1, N is set at 3, M is set at 2, and the Minicel Model takes the following special form:

$$G = C_0 + \sum_{\ell=0}^{1} h^{\ell/2} \sum_{k=0}^{1} \frac{1}{T^{(1/2+K)}} \sum_{j=1}^{3} \theta^j \sum_{i=0}^{2} C_{ijk\ell} (\ell n \sigma_s)^i \quad . \qquad \text{(IV-5)}$$

This is expanded to a 36 term polynomial as follows:

$$
\begin{aligned}
G = C_0 &+ \frac{\theta}{T^{1/2}}\left[C_{0100} + C_{1100}\,\ell n\,\sigma_s + C_{2100}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{\theta^2}{T^{1/2}}\left[C_{0200} + C_{1200}\,\ell n\,\sigma_s + C_{2200}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{\theta^3}{T^{1/2}}\left[C_{0300} + C_{1300}\,\ell n\,\sigma_s + C_{2300}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta}{T^{3/2}}\left[C_{0111} + C_{1111}\,\ell n\,\sigma_s + C_{2111}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta^2}{T^{3/2}}\left[C_{0211} + C_{1211}\,\ell n\,\sigma_s + C_{2211}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta^3}{T^{3/2}}\left[C_{0311} + C_{1311}\,\ell n\,\sigma_s + C_{2311}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta}{T^{1/2}}\left[C_{0101} + C_{1101}\,\ell n\,\sigma_s + C_{2101}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta^2}{T^{1/2}}\left[C_{0201} + C_{1201}\,\ell n\,\sigma_s + C_{2201}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{h^{1/2}\,\theta^3}{T^{1/2}}\left[C_{0301} + C_{1301}\,\ell n\,\sigma_s + C_{2301}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{\theta}{T^{3/2}}\left[C_{0110} + C_{1110}\,\ell n\,\sigma_s + C_{2110}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{\theta^2}{T^{3/2}}\left[C_{0210} + C_{1210}\,\ell n\,\sigma_s + C_{2210}\left(\ell n\,\sigma_s\right)^2\right] \\
&+ \frac{\theta^3}{T^{3/2}}\left[C_{0310} + C_{1310}\,\ell n\,\sigma_s + C_{2310}\left(\ell n\,\sigma_s\right)^2\right] .
\end{aligned}
\tag{IV-5a}
$$

Minicel Model

To finalize the Minicel Model, the MLRD program was used with Equation (IV-5), and successive analysis of variance tables were constructed as each additional variable was entered into solution. These ANOVA tables were used to determine if the incoming variable made a significant contribution to the overall correlation. A Duncan [22] F test is utilized with this format; it was found that the 25th variable that entered did not make a significant contribution to the G-level response prediction. The test hypothesis can be stated as follows:

H_0 : the entering variable has no effect upon the G-level response prediction.

H_1 : the entering variable has a significant effect upon the G-level response prediction.

The test is

$$F_{calc} = \frac{MS_{due\ to}}{MS_{about}} \qquad \text{(IV-6)}$$

and the null hypothesis H_0 can be rejected when $F_{calc} > F_{table}$. Utilizing an α level of 0.05, the final ANOVA's and F tests (Tables II and III) show that the 24th variable makes a significant contribution but the 25th does not. One further test is made to verify that all the regression coefficients in the final Minicel Model are significant. This test is conducted using a "t" statistic as follows:

$$t_n = \frac{C_n}{S_n} \qquad \text{(IV-7)}$$

where

t_n is the t statistic for the nth term

TABLE II. ANOVA TABLE AND F TESTS FOR ENTERING THE 24TH VARIABLE INTO THE MINICEL REGRESSION EQUATION

Source	S.S.	d.f.	M.S.	F_{calc}	$F_{.05}$	Decision
Due to all 24 variables	5,026,531.3	24	209,438.8	1830.76	1.52	SIGN
Due to first 23 variables	(5,024,189.8)	(23)	(218,443)	1834.9	1.52	SIGN
Due to addition of 24th variable to first 23 variables	(2341.5)	(1)	(2341.5)	20.6	3.84	SIGN
About regression on all 24 variables (Residual)	172,858.2	1511	114.4			
About regression on the first 23 variables	175,199.7	1512	115.9			
Total	5,199,389.5	1535				

TABLE III. ANOVA TABLE AND F TESTS FOR ENTERING THE 25TH VARIABLE INTO THE MINICEL REGRESSION EQUATION

Source	S.S.	d.f.	M.S.	F_{calc}	$F_{.05}$	Decision
Due to all 25 variables	5,026,771.0	25	201,070.8	1758.9	1.52	SIGN
Due to first 24 variables	(5,026,531.3)	(24)	209,438.8	1830.8	1.52	SIGN
Due to addition of 25th variable	(239.70)	(1)	239.70	2.0	3.84	NOT SIGN
About regression on all 25 variables	172,618.5	1510	114.39	2.0		
About regression on the first 24 variables	172,858.2	1511				
Total	5,199,389.5	1535				

C_n is the coefficienc of the nth term

S_n is the standard erro.: of the nth term.

The test hypothesis is as follows:

H_0 : the nth coefficient is the same as zero

H_1 : the nth coefficient is significantly different from zero.

In conducting the test, the null hypothesis can be rejected when $t_n > t_{table}$. With an α level of 0.05, the test of all the coefficients in the Minicel Model, as given in Table IV, are found to be significantly different from zero.

The resultant Minicel Model is a 25 term regression polynomial. The Minicel Model has a 0.983 correlation coefficient which compares favorably with the UAH data in Appendix A.

The reliability of the correlation coefficient can be tested statistically using a t statistic defined as follows [23]:

$$t_n = r \sqrt{\frac{n - 2}{1 - r^2}}$$

where

r = the correlation coefficient

n = the number of samples used to derive the regression equation

t_n = the resulting number of standard errors of r in the interval between the computed r and r = 0.

The test hypothesis is as follows:

$$H_0 : \quad r = 0$$

$$H_1 : \quad r > 0 \quad .$$

TABLE IV. TEST OF THE SIGNIFICANCE OF THE REGRESSION COEFFICIENTS OF THE MINICEL MODEL

Var	Coefficient Subscript Eq IV-5a	Coefficient	Stand. Error	Coef/Se	F	Decision
0	0	-.83931602E+01				
1	2100	.35419457E+01	.437855E+00	8.0893	65.4	SIGN
2	1111	-.15318724E+02	.817803E+00	-18.7316	350.8	SIGN
3	2111	.33401870E+01	.163007E+00	20.4911	419.8	SIGN
4	0101	.20749326E+03	.402348E+01	51.5707	2659.5	SIGN
5	1101	-.50350553E+02	.107843E+01	-46.6972	2180.6	SIGN
6	2101	.14344456E+01	.105041E+00	13.6561	186.4	SIGN
7	1200	-.66958791E+01	.100147E+01	-6.6860	44.7	SIGN
8	0201	-.54656687E+02	.129727E+01	-42.1322	1775.1	SIGN
9	1201	.11621323E+02	.372076E+00	31.2337	975.5	SIGN
10	0300	-.13016393E+01	.237025E+00	-5.4916	30.1	SIGN
11	1300	.20789886E+01	.140395E+00	14.8081	219.2	SIGN
12	2300	-.22664200E+00	.996818E-02	-22.7366	516.9	SIGN
13	0311	-.40141035E+00	.633909E-01	-6.3323	40.1	SIGN
14	1311	.61173036E+00	.303853E-01	20.1325	405.3	SIGN
15	2311	-.95319017E+01	.453820E-02	-21.0037	441.1	SIGN
16	0301	.39422841E+01	.126069E+00	31.2708	977.8	SIGN
17	1301	-.86603770E+00	.360774E-01	-24.0050	576.2	SIGN
18	0110	-.23377506E+03	.141474E+02	-16.5242	273.0	SIGN
19	1110	.28303458E+02	.395093E+01	7.1637	51.3	SIGN
20	0210	.49678750E+02	.386583E+01	12.8507	165.1	SIGN
21	1210	.26323240E+02	.213922E+01	12.3051	151.4	SIGN
22	2210	-.60678372E+01	.310653E+00	-19.5325	381.5	SIGN
23	1310	-.61520847E+01	.286608E+00	-21.4651	460.7	SIGN
24	2310	.10752888E+01	.509818E-01	21.0916	444.8	SIGN

In conducting the test, the null hypothesis can be rejected when $t_n > t_{table}$. This test of the reliability of the correlation coefficient was conducted on the Minicel Model. The null hypothesis can be rejected since $t_n = 291.2 > t_{table}$. The Minicel Model can be written as follows:

$$
\begin{aligned}
G = -8.39 &+ \frac{3.54\,\theta\,(\ln S)^2}{T^{1/2}} - \frac{15.31\,\theta\,h^{1/2}\,\ln S}{T^{3/2}} \\
&+ \frac{3.34\,\theta\,h^{1/2}\,(\ln S)^2}{T^{3/2}} + \frac{207.49\,\theta\,h^{1/2}}{T^{1/2}} - \frac{50.35\,\theta\,h^{1/2}\,\ln S}{T^{1/2}} \\
&+ \frac{1.43\,\theta\,h^{1/2}\,(\ln S)^2}{T^{1/2}} - \frac{6.70\,\theta^2\,\ln S}{T^{1/2}} - \frac{54.66\,\theta^2\,h^{1/2}}{T^{1/2}} \\
&+ \frac{11.62\,\theta^2\,h^{1/2}\,\ln S}{T^{1/2}} - \frac{1.30\,\theta^3}{T^{1/2}} + \frac{2.08\,\theta^3\,\ln S}{T^{1/2}} \\
&- \frac{0.23\,\theta^3\,(\ln S)^2}{T^{1/2}} - \frac{0.40\,\theta^3\,h^{1/2}}{T^{3/2}} + \frac{0.61\,\theta^3\,h^{1/2}\,\ln S}{T^{3/2}} \\
&- \frac{0.09\,\theta^3\,h^{1/2}\,(\ln S)^2}{T^{3/2}} - \frac{0.87\,\theta^3\,h^{1/2}\,\ln S}{T^{1/2}} - \frac{233.77\,\theta}{T^{3/2}} \\
&+ \frac{28.30\,\theta\,\ln S}{T^{3/2}} + \frac{49.68\,\theta^2}{T^{3/2}} + \frac{26.32\,\theta^2\,\ln S}{T^{3/2}} - \frac{6.07\,\theta^2\,(\ln S)^2}{T^{3/2}} \\
&- \frac{6.15\,\theta^3\,\ln S}{T^{3/2}} + \frac{1.07\,\theta^3\,(\ln S)^2}{T^{3/2}} + \frac{3.94\,\theta^3\,h^{1/2}}{T^{1/2}} \qquad \text{(IV-8)}
\end{aligned}
$$

where $\theta = \frac{°F + 460}{100}$ and S = static stress in psi × 100.

This model can be used to predict impact response for Minicel cushioning systems. The model is expected to be 95% reliable when used within the ranges of the independent variables which are as follows:

h = drop height from 12 through 30 inches

σ_s = static stress range from 0.03 to 5.0 psi

θ = temperature from -65° to 160°F

T = thickness of cushion from 1 through 3 inches.

The model will predict with good accuracy at all levels of the independent variables within these ranges. Also it was found that the Minicel Model does a reasonable job of predicting results beyond the ranges stipulated for the independent variables as can be seen when the results of tests of 4 and 5 inch thick Minicel samples are compared in Chapter VI.

Adjustments in Precision

The special form of the General Model as given in Equation (IV-5) was used in the validation of the model for the cross-linked polyethylene foam Minicel material and gave a correlation of 0.983. If additional precision had been required the values of S, R, M, and N could have been increased which may be necessary with other materials but gave only a minimal improvement in precision here. It was apparent, however, that increases in M, which incorporates σ_s with exponentials over 2 is in general not worthwhile. Also, increases in N above 3 that incorporate θ at exponentials of θ over 3 are of marginal value in inproved precision. The changes in R and S that affect the exponentials of thickness and drop height should be explored first if additional precision is required in a model of a particular material.

Once a special form of the General Model is found that represents a particular cushioning material, a measure of its validity can be

demonstrated utilizing the same statistical procedures as demonstrated for the Minicel Model. Improvements in the precision of the model will be reflected in increases in the "t" statistic associated with the test of the correlation coefficient. Values comparable to those of the Minicel Model are desirable.

Chapter V

VALIDATION

The model building process proceeded through many iterations of development and verification and culminated in the General Model of impact response stated mathematically as

$$G = C_0 + \sum_{\ell=0}^{S} h^{\ell/2} \sum_{k=0}^{R} \frac{1}{T^{(1/2+K)}} \sum_{j=1}^{N} \theta^j \sum_{i=0}^{M} C_{ijk\ell} (\ell n\ \sigma_s)^i \ . \qquad \text{(V-1)}$$

This gives the basic underlying structure for a model of impact response for bulk cushioning materials. The question of how good a model has been developed is answered in the validation process. Naylor and Finger [24] suggest three stages of validation:

1) Verification of internal structure

2) Empirical testing

3) Verification as a predictor.

Verification of Internal Structure

The validating process began when the first model of impact response was developed. At that time the ingredients of a model were selected and their relationships postulated on the basis of prior knowledge and existing theory. The basis of the General Model was founded in the theory of viscoelasticity, and the individual parameters (θ, σ_s, T, h, G) were arranged in the model in a manner that is consistent with theory and intuition. The General Model that resulted from

the modeling effort incorporates the following characteristics:

1) The dynamic cushioning curves of G-level versus static stress generated by the model are U shaped. This is consistent with cushioning theory.

2) The predicted G-levels increase with increased drop height. The higher drop heights incorporate more energy into the system which would increase the energy levels experienced by the falling body.

3) The predicted G-levels increase with reduced cushion thickness. The G's experienced by a falling body is dependent on shock pulse duration, Equation (II-1), and a thinner cushion would allow less shock pulse time and an accompanying increase in G-levels.

4) Temperature effects induce lateral shifts in the dynamic cushioning curves. This is consistent with the phase shift function concept of viscoelastic theory. It is also intuitively consistent in that reduced temperatures would be expected to stiffen the cushioning material and require an increased stress level for comparable shock attenuation.

5) The dynamic cushioning curves generated as output from the General Model form a series of curves that are nested within progressive values of thickness and drop height for all possible temperature conditions and static stress conditions. It can be concluded that the internal structure of the General Model is intuitively correct and the output of the General Model is consistent with expectation.

Empirical Testing

The General Model of Equation (V-1) is hypothesized as the model of impact response that is applicable as the basic underlying structure

of a model for any one of the many cushioning materials. An impact response model for any one particular cushioning material can be constructed by establishing the summation levels M, N, R, and S, and the values of the regression coefficients in the General Model. This can be done through a testing program that generates a data base similar to Appendix A. Then an analysis program is required that provides a least squares fit of the data base. The test program required can be similar to the one conducted in the UAH study. Once the test program is completed and the data base established, an analysis must be performed to develop the model. The stepwise regression analysis given in Appendix B or a similar analysis routine can be utilized.

When this procedure was followed in constructing the Minicel Model, Equation (IV-8), the General Model was used as the basic underlying structure and the statistical tests performed in verifying the Minicel Model serve to validate the General Model. The ANOVA Table for the Minicel Model (Table II), showing an F_{calc} of 1830.76 and a correlation coefficient of 0.983, is indicative of the fit of the Minicel Model to the UAH results. In addition to the high correlation of the model with experimental data, the Minicel Model also demonstrates the characteristic U shaped dynamic cushioning curves which were one of the more important aspects of the impact response model considered in the model development. The dynamic cushioning curves produced from the Minicel Model were plotted using the MLRD printer plot routine for many conditions of thickness, drop height, and temperature. The classic U shape was evident in all the plots, six of which are given in Appendix E.

An additional measure of validity of the model can be seen when the best fitting polynomials in the UAH study are compared to the Minicel Model. The Minicel Model plots in Appendix E have the independent variables at the same levels as the plots of the UAH curves in Appendix C.

Verification as a Predictor

The final test of validity of the General Model is to assess the ability of the model to predict impact response. It was previously demonstrated that the Minicel Model does an excellent job of predicting impact response when compared to the actual data. However, it must be remembered that the Minicel Model is dependent upon these data when formulating its predictions. It is a limited dependency in that it is using all 2709 data points in predicting a particular dynamic cushioning curve, and in fact only a small portion of the data base, approximately 50 data points, apply directly to the particular conditions with the independent variables at the appropriate levels. The UAH best fitting polynomials, however, are derived using only those data points where the independent variables are at the appropriate levels.

To fully verify the model as a predictor, a data base independent of that used to generate the model must be utilized. Three such data bases are available: a data base of 1, 2, and 3 inch thickness Minicel material at -65°, 70°, and 160°F and at 27 inch drop height, and a data base of 4 and 5 inch Minicel material. The 27 inch drop height data are given in Appendix F and the 4 and 5 inch data in Appendix G. The 27 inch data are contained within the range of the independent variables and was not used in formulating the Minicel Model. However, the 4 and

5 inch data are beyond the data extremes of the developed model. Also the 4 and 5 inch samples themselves were not homogeneous. The samples of the 4 inch material were two-piece cushions which were 2 inches thick. The 5 inch material was made using a 2 inch and a 3 inch piece. This stacking is representative of how cushioning is actually used in shock isolation systems requiring thicker sections than the maximum manufactured thickness of 3 inches. Whether the Minicel Model can do an adequate job of predicting the impact performance of these stacked samples is questionable. The model's ability to predict adequately under these circumstances would indicate that the stacking did not significantly perturb the cushioning performance from that encountered in the 1, 2, and 3 inch continuous samples that form the basis of the model.

To determine statistically how well the model fits a set of independent data, Box and Draper [25] suggest it is appropriate to investigate the bias and variance of the predictor. Two statistical tests can be utilized for this purpose. One test, a test of means, determines whether there is a bias in the predicted values of the model when compared to actual data values. The other test, a test of variances, determines if the variations of the predicted values inherent in the model are comparable with the variations in experimental values.

Test of Means

In all the data bases, including the UAH data, there are three replications of the same conditions of the independent variables. These three replications can be considered a cell. Then, the mean of the G-levels of the three data values in a cell (G_i, i = 1, 2, 3) can be compared with the predicted G-level from the model for that cell, and

it is reasonable to expect the sum of the differences to vanish. Any difference that cannot be reasonably attributed to sampling error must be considered a bias that is introduced because the model values are not good predictors.

The first step in this test is to formulate a difference between the data values in a cell and the model prediction for that cell. This can be expressed mathematically as follows:

$$\Delta_j = (GM_j - \overline{GD}_j) \qquad \text{(V-2)}$$

where

j = number of a cell with fixed values of θ, σ_s, T, h

Δ_j = cell difference for cell j

GM_j = the G-level predicted by the model for cell j

$\overline{GD}_j$ = the mean value of G-level for the three data values in cell j

$$\overline{GD}_j = \frac{\sum_{i=1}^{3} G_i}{3} .$$

Then S, the standard deviation of all the cells, can be defined as follows:

$$S = \sqrt{\frac{\sum_{j=1}^{N} (\Delta_j - \bar{\bar{G}})^2}{N - 1}} \qquad \text{(V-3)}$$

where

N = the number of cells in the data base

$\bar{\bar{G}}$ = the grand mean of all the cell differences, $\bar{\bar{G}} = \frac{\sum_{j=1}^{N} \Delta_j}{N}$.

The hypothesis to be tested is based on the expected value of the differences which can be written $E(\triangle_j)$ and stated as follows:

$$H_0: \; E(\triangle_j) = 0$$

$$H_1: \; E(\triangle_j) \neq 0 \quad .$$

A two tailed "t" test is used where the t statistic is given as

$$t_m = \frac{\bar{\bar{G}} \sqrt{N}}{S} \tag{V-4}$$

where t_m = the test statistic for the model.

The test compares t_m with t_{table} and the null hypothesis can be rejected when $t_m > t_{table}$. Rejection of the null hypothesis implies that $E(\triangle_j)$ is sufficiently greater than zero as to be unexplainable as sampling error and therefore the model predictions would appear not to be representative of the data.

Test of Variances

The other test of goodness of fit of the Minicel Model with actual data determines how the variations in the prediction, using the model, compare with the variations in the data. In each data base, the data points in each cell are the replications of the experiment for each set of conditions of stress level, drop height, and temperature. The sum of these variations for each cell can be written as follows:

$$T_j = \sum_{i=1}^{R} (G_i - \overline{GD}_j)^2 \tag{V-5}$$

where

i = the number of the sample in the cell being considered

T_j = the sum of squares of the variation for cell j

R = the number of replications in each cell

G_i = G-level value of the data point being considered

$\overline{GD}_j$ = mean value of the data values in the cell being considered from Equation (V-2).

Then σ_d^2 can be defined as the data within-cells variance which is found by summing the variations within all the cells:

$$\sigma_d^2 = \frac{\sum_{j=1}^{N} T_j}{N \times df} \qquad \text{(V-6)}$$

where df = degrees of freedom in each cell.

The hypothesis to be tested is whether the variance of the data base is equal to the variance in the model predictions.

The test hypothesis can be expressed as follows:

$$H_0 : \sigma_d^2 = \sigma_m^2$$

$$H_1 : \sigma_d^2 \neq \sigma_m^2 \quad .$$

where

σ_d^2 = the variance in the data as given in Equation (V-6)

σ_m^2 = the variance in the predicted values from the model.

An "F" value is used to test the ratio of variances and the F statistic is defined as follows:

$$F_{calc} = \frac{\sigma_m^2}{\sigma_d^2} \qquad \text{(V-7)}$$

The model variance is computed during the regression procedure as the Residual Mean Square in the analysis of variance of the

model being tested. The test compares F_{calc} with F_{table}. F_{calc} is computed from Equation (V-7) and F_{table} is set using $\alpha = 0.05$ and is tabled according to the degrees of freedom in the numerator (the degrees of freedom in the data base being considered) and the degrees of freedom of the demoninator (the degrees of freedom of the model being tested).

Rejection of the null hypothesis indicates that there is a difference between the model variance, σ_d^2, and the data variance, σ_m^2, that cannot be attributed to sampling error. This implies that there is a significant difference in the variance of the model when compared with the actual data and that the model is not representative of the data.

Prediction Test Results

The test of means and variances were conducted on the Minicel Model, Equation (IV-8), using the 27-inch drop height data in Appendix F and the 4 and 5 inch data samples of Appendix G. The results are given in Table V and show the following:

1) 27-inch drop height data - The 27-inch drop height data are within the range of the independent variables used in the Minicel Model and the null hypothesis cannot be rejected in the test of means or test of variances. This appears to indicate that there is no significant difference between the means and variances of the data and the values predicted by the model. The model appears to be a statistically valid predictor of these data.

2) 4 and 5 inch data - The test results for the 4 and 5 inch data show that the Minicel Model gives good predictions of the

TABLE V. TESTS OF MINICEL MODEL AS A PREDICTOR OF 4 AND 5 INCH THICKNESS AND 27 INCH DROP HEIGHT

	Minicel Material (α = 0.05)		
	27 inch Drop Height	4 inch Thickness	5 inch Thickness
Test of Means			
Number of cells in the data base (N)	99	156	156
Standard deviation of the cells (S)	20.14	8.63	8.56
t_m	1.84	-1.45	-2.61
t_{table}	±1.96	±1.96	±1.96
Decision on H_0 : $E(\Delta_j) = 0$	cannot reject	cannot reject	reject
Test of Variances			
Variance of the data, σ_d^2	111.63	26.41	105.35
Variance of the model, σ_m^2 (Residual Mean Square, Table IV)	114.39	114.39	114.39
F_{calc}	1.02	4.33	1.08
F_{table}	$F_{1511,198} = 1.17$	$F_{1511,312} = 1.15$	$F_{1511,312} = 1.15$
Decision on H_0 : $\sigma_d^2 = \sigma_m^2$	cannot reject	reject	cannot reject

data means at the 4 inch thickness. The test of means on the 4 inch showed no significant difference but the test on the 5 inch showed a significant difference which is not surprising because the model is projecting 2 inches beyond its range at the 5 inch thickness. The test of variances shows that the model variance at the 4 inch thickness is significantly larger, but this is not true at the 5 inch level. The model dispersion is comparable to the dispersion of the 5 inch data.

There were reservations as to whether the model could predict the 4 and 5 inch materials since the samples were stacked and not homogenous and since 4 and 5 inches are beyond the ranges of the independent variables used in the Minicel Model. The results of the tests of means and variances show that the model was not completely acceptable at either the 4 inch or 5 inch thickness. However, the tests indicate that model means were comparable with the data at the 4 inch level and the variances were comparable at the 5 inch level.

Chapter VI

OPTIMIZATION

The procedure which was designed in this research to perform cushion system optimization can best be described as a constrained sequential search technique. The technique uses the knowledge that all dynamic cushioning curves are U shaped, and searches for limits of acceptable G-level values along these curves. The procedure is a direct search technique in that the mathematical model of a material such as the Minicel Model, Equation (IV-8), is used directly as the objective function in the optimization procedure.

The first step in the optimization exercise is to identify the parameters in an optimal cushioning system design; then, the objective or goal of the optimization procedure must be identified. The constraints on the procedure and the objective function are formulated and finally the outputs from the optimization routine itself are identified.

Formatting for Optimal Cushion System Design

An optimal cushioning system is a system that provides the necessary shock isolation to the protected item at a minimum cost. Because the cost of a cushioning system is dependent upon the amount of cushioning material, the optimal cushioning system will employ the minimum thickness of cushion needed. The optimal point on a dynamic cushioning

curve such as that shown in Figure 10 is the minimum G's possible which provides the maximum protection for that particular thickness of cushioning. Therefore, it is the identification of this optimal point that must be determined in selecting the optimal cushion system.

A different dynamic cushioning curve is required to depict cushion performance for each drop height, thickness, and temperature of the cushion; the optimal point is different for each curve. When one considers the many types of cushioning materials and a different curve for each condition, it can be seen that a very large library of curves is required to present even the most likely conditions of drop height, temperature, and thickness of a few candidate cushioning materials.

A model of impact response for a cushioning material, e.g. the Minicel Model, computes the impact response directly from the values of the independent variables. This precludes the need for a library of dynamic cushioning curves in predicting impact response. It is also possible, using the cushioning models for various materials, such as the Minicel Model, to determine an optimal cushioning design for each material, if such a design exists. It is also important to provide the designers with the maximum amount of useful design information in the output.

Specification of the Cushion System Constraints

The development of a valid mathematical model of impact response for a particular cushioning material provides a vehicle for the application of optimization techniques. This model can be used as an objective function and explored for an optimal cushioning system. This can be done by defining cushioning system design requirements in terms of the external environment and the amount of exposure the protected item can withstand.

The specification of the external environment must contain a definition of those aspects of the environment that have an impact upon cushion system design. This would include a quantitative measure of the magnitude of the maximum shock pulse to which the system will be exposed, which is usually specified in terms of drop height. Also since temperature has a significant effect on impact response, the specification of the external environment should include the range of temperatures to which the system will be exposed (θ_{min}, θ_{max}).

The specification of the survivability of the protected item is given in terms of its ability to withstand shock. The maximum shock pulse, in G's, that the item can withstand is given as the fragility level of the item (GL_{max}).

These considerations concerning the G-levels and temperatures are incorporated into the model as input values that are utilized to constrain the optimal search to feasible solutions.

Construction of the Objective Function

Consider the parameters of the general cushion design model that can be expressed mathematically in functional notation as

$$G = F(\sigma_s, T, \theta, h) \quad . \qquad \text{(VI-1)}$$

For the purpose of optimal design, GL is identified as the fragility level of the protected item. The temperature parameter, θ, must consider the range of temperatures of the external environment θ_{min} to θ_{max}. The superimposed dynamic cushioning curve format, shown in Figure 2, can be used for this purpose wherein the curves of extremes of the temperature range are superimposed upon the ambient curve.

One parameter in the model, drop height (h), can be determined based on published testing standards. Other parameters, such as the fragility level of the item to be protected (GL_{max}) and the temperature range (θ_{min}, θ_{max}), are usually specified by overall system requirements. For example, if the cushioning system is to be designed for a military container for a missile, the fragility level of the protected item (GL_{max}) will be specified as part of the hardware specifications in the missile system design criteria. The temperature range (θ_{min}, θ_{max}) is defined in the missile system requirements and the drop height is specified in various Military Standards depending on total container weight. These parameters, GL_{max}, θ_{min}, θ_{max}, and h, are the exogenous variables in the optimization model. Equation (VI-1) can now be written as follows:

$$GL_{max} = F[\sigma_s, T, (\theta_{min}, \theta_{max}), h] \qquad \text{(VI-2)}$$

where GL_{max}, (θ_{min}, θ_{max}), and h are inputs to the equation which are determined by the cushion system design requirements. The optimization technique will search out the optimal design from this expression, and the optimal design solution will be expressed in terms of σ_s and T.

The optimal design searches upon the functional relationship expressed in Equation (VI-2) and with the inputs introduced, reduces to the selection of the minimum thickness of cushion that will perform satisfactorily at a static stress condition that is determined in the search. Classical search techniques would establish Equation (VI-2) as the model to use in the search procedure and it would be rearranged with T as the dependent variable and the objective function as follows:

$$T_{min} = F[\sigma_s, GL_{max}, \theta, h] \quad . \qquad \text{(VI-3)}$$

The search would be subject to certain constraints such as $[\theta_{min} < \theta < \theta_{max}]$ and $[GL \leq GL_{max}]$, and a search made for T_{min} using one of many search techniques.

However, it is not apparent how Equation (VI-2) can be rearranged mathematically into the simple form of Equation (VI-3) where T is solved for directly, nor is it necessarily desirable to do so. It is important to recognize that when the exogenous variables are introduced into the objective function, a three-dimensional search on σ_s and T is all that is required. Since the general shape of the curve of G versus σ_s is known, i.e., it is a classical U shaped dynamic cushioning curve, Equation (VI-2) can be searched directly with little difficulty and with reasonable efficiency.

Once Equation (VI-1) is rewritten as Equation (VI-2) with the values of h, $(\theta_{min}\ \theta_{max})$, and GL_{max} introduced, we have an objective function that can be written as follows:

$$GL_{max} = F\left(\sigma_s, T, (\theta_{min}, \theta_{max}), h\right)$$

where GL_{max}, θ_{min}, θ_{max}, and h are input constraints. This objective function is then searched using the direct search routine.

The Direct Search Routine

The direct search routine (CUSHION OPT) is a three stage process. The initial stage involves the selection of a material type. It is anticipated that a data base will eventually be available that contains valid impact response models for many of the different types of bulk cushioning materials. The model of each type of material in the data

base can be investigated by the direct search routine to determine the feasibility of meeting the design objective function. The optimal design conditions of minimum thickness and acceptable static stress range will be output for each type of cushion in the data base if an optimal design exists.

The initial step is to select the type of material. Then, a search is initiated on a minimum thickness cushion (1 inch) at the drop height and minimum temperature (θ_{min}) stipulated in the design requirements. These values are input into the objective function and the search is conducted across the static stress spectrum at the temperature extreme, θ_{min} and then θ_{max}, to determine the feasibility of meeting the stipulated GL_{max}. It is also necessary that the acceptable static stress range at GL_{max} be greater than 0.2 psi to permit design flexibility and preclude creep problems within the design. (This 0.2 psi value can be changed at the discretion of the designer.).

If the search of the acceptable static stress range on the minimum cushion thickness is successful, the answer is output as a feasible solution. If it is not successful, the thickness is incremented 1/2 inch and the search repeated. A flow chart of the CUSHION OPT search routine is given in Figure 14 and the program code is given in Appendix H.

Results

The CUSHION OPT Program output is given in the form of superimposed dynamic cushioning curves such as the typical one given in Figure 15. In Figure 15, the design objective function had the temperature range of -65° to 160°F and a 30-inch drop height. The fragility level was

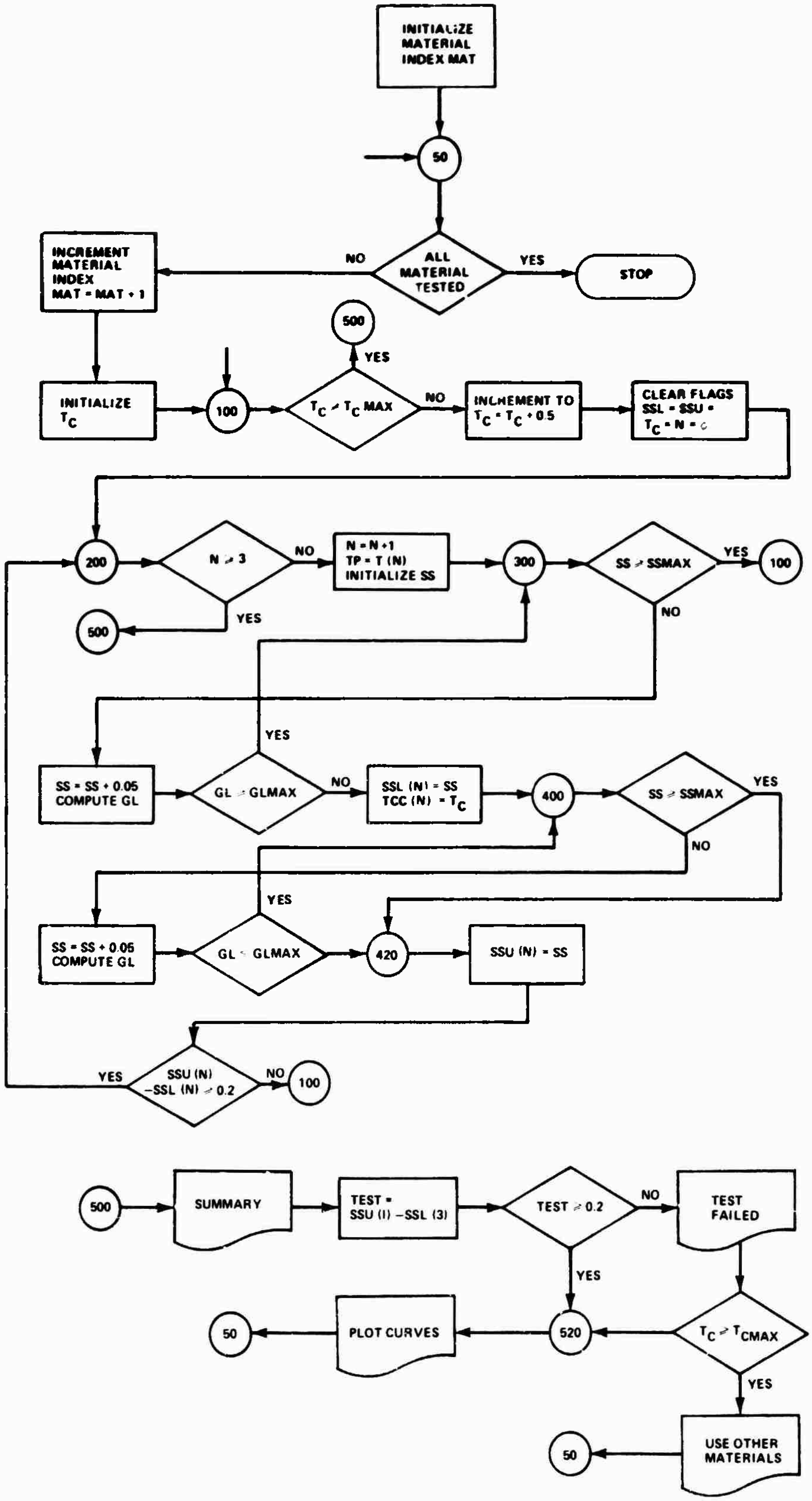


Figure 14. Flow chart of optimization direct search routine (CUSHIONOPT).

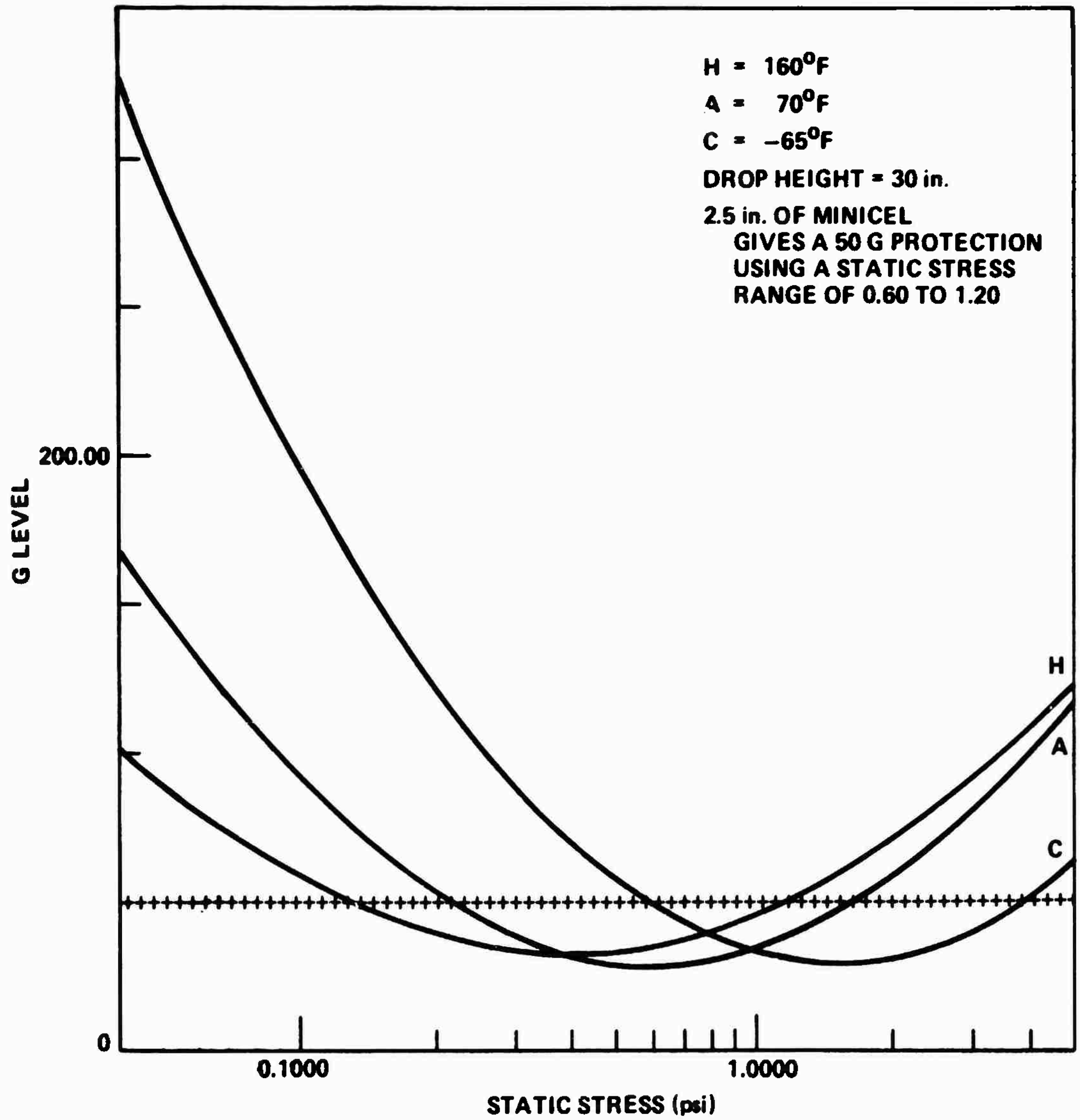


Figure 15. Optimal design output, 2-1/2 in. Minicel.

50 G's which is indicated with a dotted horizontal line. The optimal thickness is determined to be 2.5 inches and the feasible static stress range is found to be 0.60 to 1.20 psi. One of the inherent advantages of using math models is seen here in the capability of determining G-level response at thicknesses other than those in the data base. In this instance a 2.5-inch thickness of cushion was optimal; tests were not conducted at this thickness in the data base. This advantage can also be seen in Figure 16 which is another output of the optimization program where the temperature range has the non-data base values of -20° and 120°F. The optimal thickness is 2.0 inches and the stress range is 0.55 to 0.85 psi to give 50 G protection. Comparison of Figure 15 with Figure 16 demonstrates the effect of temperature on optimal cushion design. When the temperature range was relaxed from the extremes of -65° through 160°F to -20° through 120°F and all other exogenous variables kept the same, the thickness of cushion required for 50 G protection dropped from 2.5 inches to 2 inches. Another comparison can be made between Figure 15 and Figure 17, which demonstrates the increase in thickness of cushion required when the fragility level of the protected item is lowered from 50 G's to 40 G's. One other comparison can be made between Figure 15 and Figure 18, which demonstrates how a reduced drop height requirement reduces the thickness of cushion required.

One additional advantage in using the superimposed dynamic cushioning curve format for the output format is that the designer is presented with a convenient tool for minimizing the creep tendency of the materials by maintaining as low a stress condition as possible and yet be able to ascertain the response of the stress level selected.

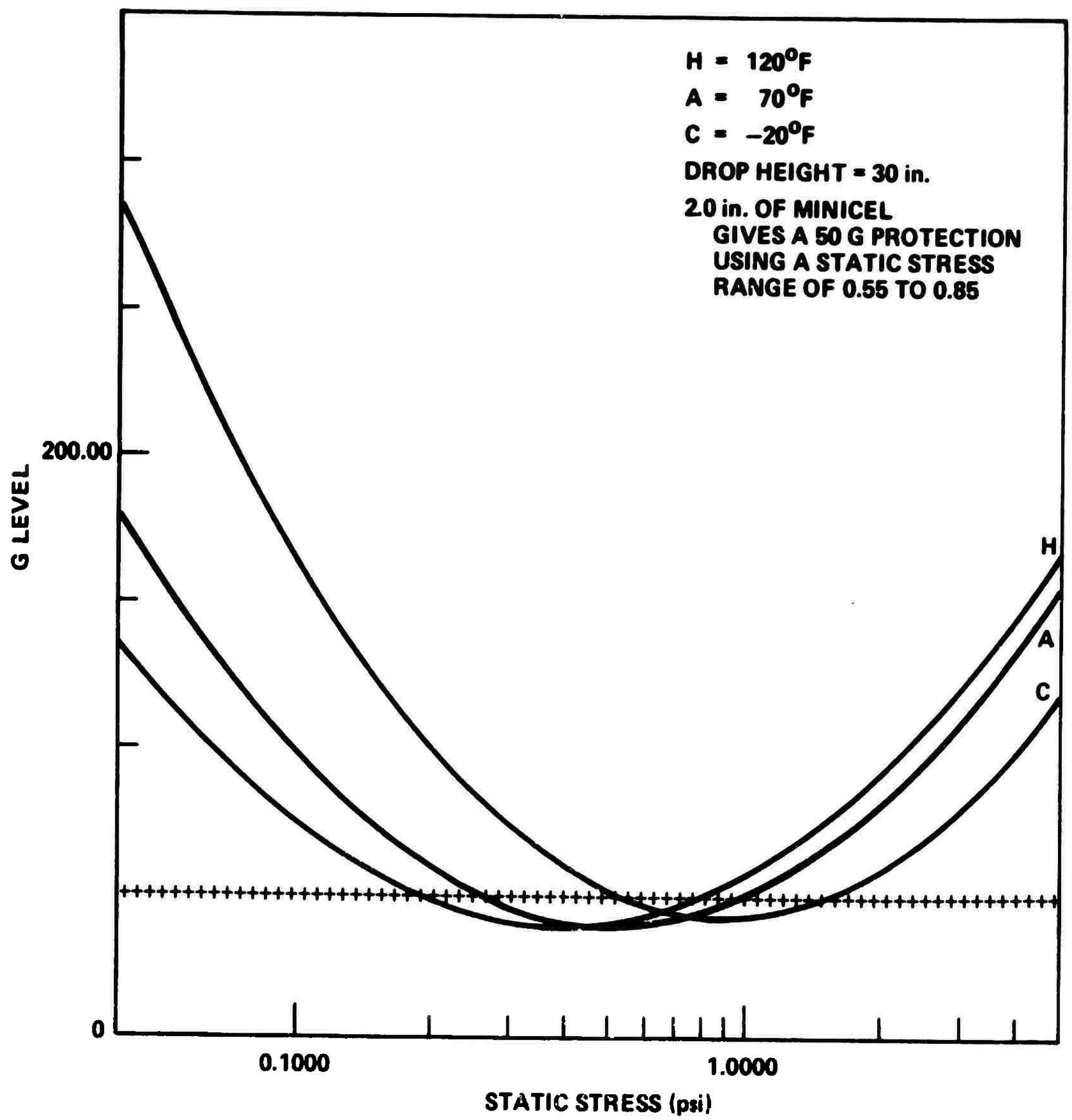


Figure 16. Optimal design output, 2 in. Minicel.

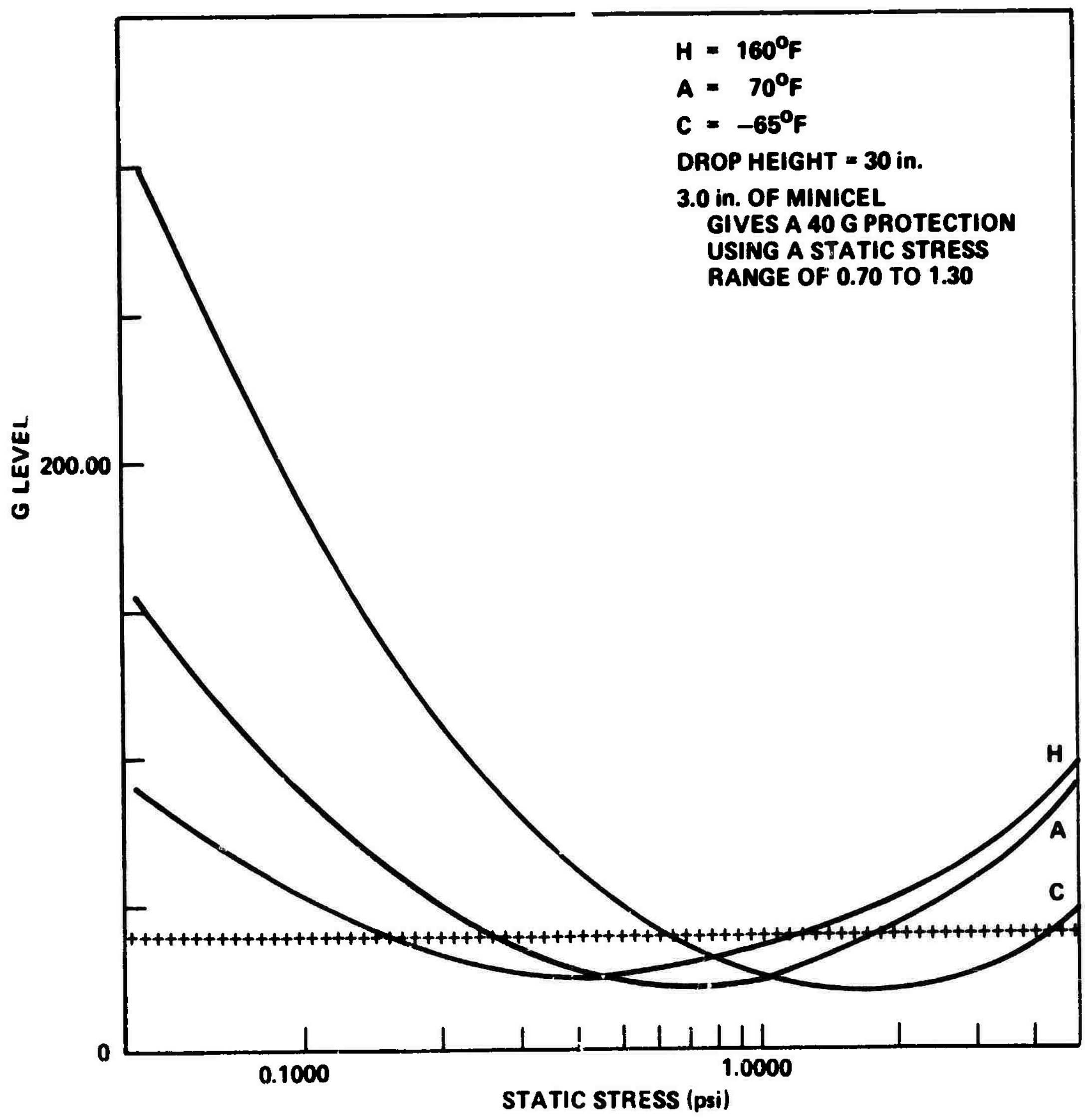


Figure 17. Optimal design output, 3 in. Minicel.

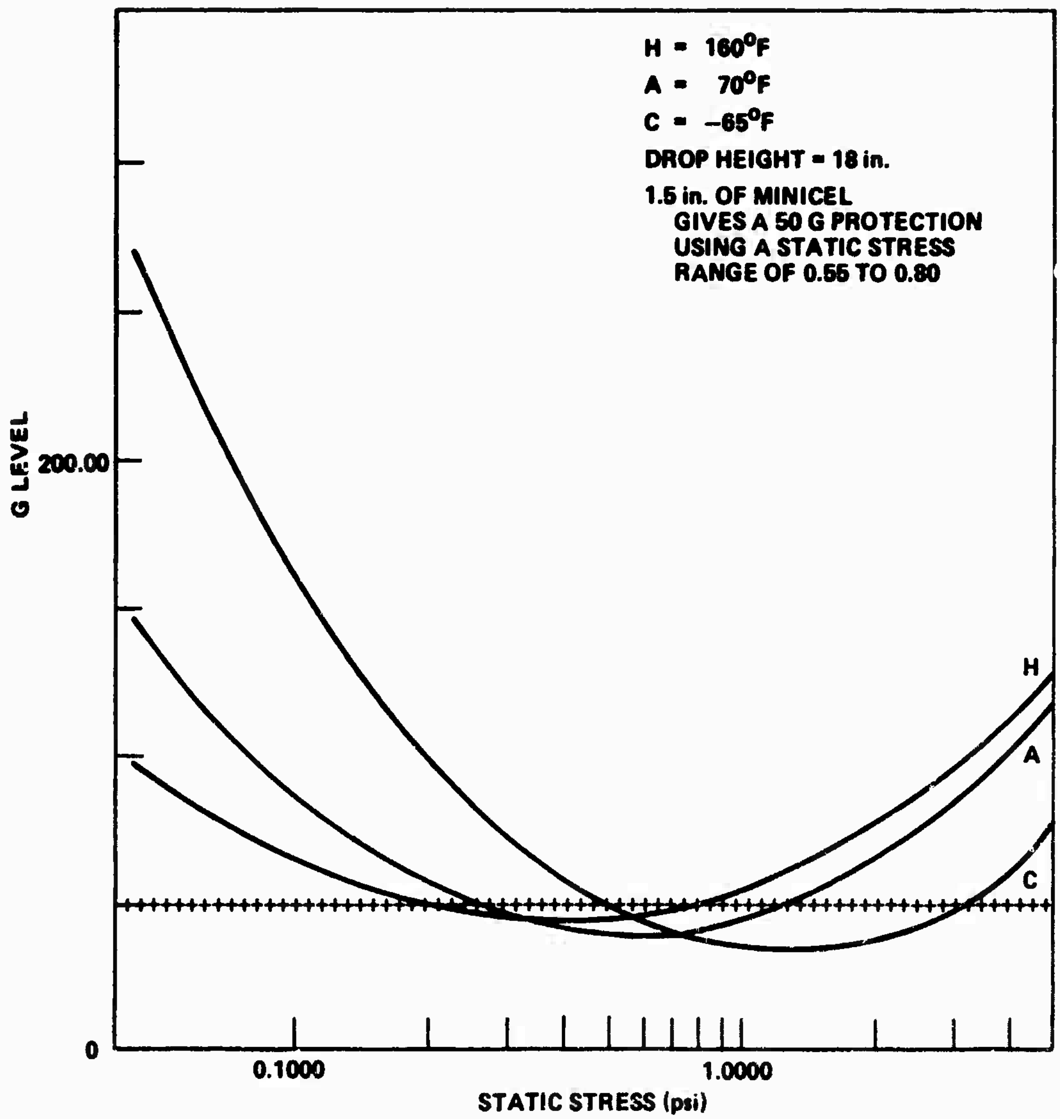


Figure 18. Optimal design output, 1-1/2 in. Minicel.

Chapter VII

CONCLUSIONS AND RECOMMENDATIONS

There has been limited use made of bulk cushioning in shock isolation systems for commercial and military equipment. One reservation that designers have had regarding bulk cushioning is an inability to predict cushioning performance at temperature extremes. Consequently, many designs incorporate mechanical suspension systems such as elastomeric mounts or springs and dash pots and other types of shock isolation systems that are bulky, very heavy, require substantial structural interfaces, and are very expensive.

One result of this research is the development of a valid model of bulk cushioning response that provides a basis for improving the predictability of temperature performance for bulk cushioning materials. Resultant increases in the use of bulk cushioning for shock mitigation systems will generate dollar savings in the accompanying reduced procurement and development costs.

Conclusions

The developed General Model of the impact response of bulk cushioning materials is

$$G = C_0 + \sum_{\ell=0}^{S} h^{\ell/2} \sum_{k=0}^{R} \frac{1}{T^{(1/2+K)}} \sum_{j=1}^{N} \theta^j \sum_{i=1}^{M} C_{ijk\ell}(\ell n\, \sigma_s)^i \quad . \tag{VII-1}$$

The model is predicated on viscoelastic theory and incorporates the effect of temperature, stress, drop height, and thickness of cushion upon the impact response of a cushioning system. This General Model provides the basic underlying structure of impact response of any one of the many bulk cushioning materials used for shock isolation. A sensitivity analysis can be run on the values of S, R, N, and M to obtain the precision desired for an impact model of a particular cushioning material.

Models that are predicated on the basic underlying structure of the General Model are better predictors of impact response than the dynamic cushioning curves currently being utilized, because the effect of temperature has been incorporated into the model. The Minicel Model is one such model that was constructed for the Hercules, Inc. 2 lb/ft^3 Minicel material using the General Model as the basic underlying structure. The Minicel Model is a 25-term polynomial given in Equation (IV-8). The correlation of the Minicel Model with actual data demonstrates the validity of the models and their value as predictors of impact response. The Minicel Model showed high correlation with the actual data within the ranges of the variables and also showed promise as a predictor beyond the variable ranges.

The development of a valid model of impact response of bulk cushioning materials defines the relationships and interrelationships of the variables in response to impact. Using this basis, it was possible to employ a search technique (CUSHION OPT) to determine optimal cushion design and display the findings in terms of superimposed dynamic cushioning curves. The functional form of the model provides the advantage of determining impact response at non-tested levels of the

variables with confidence and precludes the need for a library of dynamic cushioning curves for all the different combinations of conditions.

Once a valid model of a particular cushioning material has been developed, it can be incorporated into the CUSHION OPT optimization program. This program accepts the design requirements for a shock isolation system and computes and provides, in the form of superimposed dynamic cushioning curves, the optimal design for each cushioning material in the data base, if one exists. The outputted superimposed dynamic cushioning curves give the pertinent information needed in the optimal design of a shock isolation system.

Recommendations

It is recommended that the model of impact response of bulk cushioning materials, Equation (VII-1), be used as the basic underlying structural design for cushioning systems. This model is considerably better than any previous basis of design. The optimization program used in conjunction with the model can be used to provide accurate predictions of shock mitigation system performance in a time saving manner and in a useful format. It is reasonable to expect that in the design of shock isolation systems, considerable savings can be realized in design time and cost savings by using these more accurate response predictions prior to prototype fabrication and test.

Further, it is recommended that the optimal design program, CUSHION OPT, be used to generate the optimal design of bulk cushioning systems. Cushioning system designers that have a large scale digital computer capability should be encouraged to utilize this procedure. As models

of additional cushioning materials become available, CUSHION OPT is provisioned to permit updating to incorporate additional design alternatives using the new materials. To insure maximum utilization of the improved predictive capability of the General Model, it is further recommended that superimposed dynamic cushioning curves of the most likely conditions of the independent variables ([illegible], σ_s, h, T) be published and made available to cushioning system designers who do not have access to a computer facility. It is recognized that one of the major advantages inherent in the CUSHION OPT program, that of obtaining the exact design constraints needed, becomes inoperative and therefore, manual interpolation will be necessary when using published curves. However, the difficulties associated with using and maintaining a library of superimposed dynamic cushioning curves are warranted when the savings accrued through the use of the improved predictions are considered.

Additional drop test programs, similar to the one conducted on Minicel, should be conducted on the bulk cushioning materials. A model of impact response similar to the Minicel Model, Equation (IV-8), using the General Model as the basic underlying structure should be constructed for each new material. This model can then be entered into the CUSHION OPT program to provide an additional optimal design alternative in terms of this new material. It is further recommended that the drop test programs on additional materials be conducted using levels of the independent variables ([illegible], σ_s, h, T) that bracket the values of the variables that might occur in cushion system design. This increase in the range of the variables would insure that the model is predicting values within the range of the model. Particular attention should be

given to insuring that the range of thickness is sufficient to obtain G-levels as low as 10 G's, which is not uncommon in some of the more fragile optical and electronic hardware. Also, since only the lower portions of dynamic cushioning curves are used in optimizing cushion design, the test programs can be abbreviated in the understressed and overstressed regions.

It is also recommended that the General Model of impact response be considered as a basis for future research. The mathematical formulation of impact response should provide a vehicle to be utilized for the rigorous analysis of impact response. For example, one particularly lucrative area is that the model be used as a phenomonological constitutive equation in advancing the viscoelastic theory of bulk cushioning materials.

REFERENCES

1. McDaniel, Don, A Study of Impact Test Effects Upon Foamed Plastic Containers, US Army Missile Command, Redstone Arsenal, Alabama, Report No. RL-TR-71-2, 1971.

2. US Army Rocket and Missile Container Engineering Guide, US Army Missile Command, Redstone Arsenal, Alabama, June 1971.

3. Mindlin, R. D., "Dynamics of Package Cushioning," Bell System Technical Journal, 24, and Bell System Monograph B-1369, July-October 1945.

4. Gigliotti, M. E., "Design Criteria for Plastic Package-Cushioning Materials," PLASTEC, Picatinny Arsenal, Plastics Technology Evaluation Center, Dover, New Jersey, Report 4, December 1961.

5. Stern, R. K., The Cushion Factor Stress Curve and its Value for Classifying and Selecting Package Cushioning Materials, Wright Air Development Center, Dayton, Ohio, Technical Report 58-233, November 1958.

6. Franklin, P. E. and Hatae, M. T., "Packaging Design," Shock and Vibration Handbook (C. M. Harris and C. E. Crede, eds.) McGraw-Hill, New York, 1961.

7. Krakover, S. M. and Olevitch, A., Investigation of Design Criteria for Cushioning Materials, Wright Air Development Center, Technical Report WADC-TR-58-639, November 1958.

8. Properties of Latex Hair Package Cushioning Material for Shock and Vibration Protection, North American Aviation, Design Specifications Group, Report AL-2609, 9 September 1957.

9. Janssen, R. R., A Method for the Proper Selection of a Package Cushion Material and its Dimensions, North American Aviation, Los Angeles, California, Report NA-51-1004, 1952.

10. Kerstner, O. S., General Principles of Package Design, Part I - Cushioning, Northrop Aircraft, Inc., Hawthorne, California, Report NAI-57-187, February 1957.

11. Humbert, W. G. and Hanlon, R. G., Package Engineering, "Acceleration Versus Static-Stress Curves for Polyethylene Foam," April 1962.

12. Mustin, Gordon S., Theory and Practice of Cushion Design, The Shock and Vibration Information Center, United States Department of Defense, 1968.

13. Wyskida, Richard M. and Wilhelm, Mickey R., Temperature Sensitive Dynamic Cushioning Function Development and Validation for Hercules Minicel Thermoplastic Foam, The University of Alabama in Huntsville, Huntsville, Alabama, UAH Research Report No. 159 (MICOM Report No. RL-CR-75-1), September 1974.

14. Sharma, M. G., Viscoelastic Behavior of Plastics, Summer Institute on Applied Mechanics and Materials Science, The Pennsylvania State University, University Park, Pennsylvania, 9-14 June 1963.

15. Meinecke, E. A. and Clark, R. C., Mechanical Properties of Polymeric Foams, Technomic Publishing Co., Inc., Westport, Connecticut.

16. Tobolsky, A. V., "Stress Relaxation Studies of the Viscoelastic Properties of Polymers," Rheology, Theory and Applications, Vol. 2 (F. R. Eirich, ed.) Academic Press, New York, 1958.

17. Cost, Thomas L., Dynamic Response of Container and Container Cushion Structures, Athena Engineering Company, Northport, Alabama, Technical Report No. 74-003, October 1974.

18. Draper, N. R. and Smith, H., Applied Regression Analysis, John Wiley & Sons, Inc., New York, 1966.

19. Gent, A. N. and Thomas, A. G., "Mechanics of Foamed Elastic Materials," Rubber Chem. and Tech., 36, 1963.

20. Mechlin, G. F., "Polyurethane Foams for Shock Mitigation Systems," Modern Plastics, August 1966.

21. Shannon, R. E., Systems Simulation: The Art and Science, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1975.

22. Duncan, Acheson J., Quality Control and Industrial Statistics, Richard D. Irwin, Inc., Homewood, Illinois, 1965.

23. Daniel, Bailey E., Probability and Statistics - Models for Research, John Wiley & Sons, Inc., New York, 1971.

24. Naylor, Thomas H. and Finger, J. M., "Verification of Computer Simulation Models," Management Science, Vol. 14, No. 2, October 1967.

25. Box, G. E. P. and Draper, N. R., "A Basis for the Selection of a Response Surface Design," Journal of American Statistical Association, Vol 54, 1959.

Appendix A

UAH DATA, 12, 18, 24 AND 30 INCH

MINICEL - 12 in. Drop Height

STRESS LEVELS (PSI)

	0.04			0.10			0.20			0.40			0.80			1.0			1.6			2.0			
	Temperature (°F)																								
Replication	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	251*	152	103	148	72	58	89	47	45	44	45	47	37	39	49	34	46	52	31	46	49	35	60	70	1"
2	205	150	108	141	72	56	91	49	49	50	43	43	37	42	45	30	42	48	29	54*	55	34	57	67	
3	175	158*	103	156	75	55	78	49	44	53	42	46	37	43	45	34	41	52	32	45	57	33	61	66	
1	208	134*	84	139	78*	43	69	34	28	44	30	25	32*	25	25	24	23	25	21	21	23	18	22	28	2"
2	206	117	82	148	55	42	81*	34	29	43	29	26	25	23	24	26	22	23	18	22	23	17	24	28	
3	209	119	79	128*	58	40	72	35	29	45	27	25	28	22	25	24	22	24	18	19	22	17	25	25	
1	155*	109*	72	94	50	37	63	30	26	40	21	20	25	15	20	22	21	24	15	21	24	16	14	17	3"
2	217	102	61	108	47	36	64	31	23	42	22	20	24	19	15	20	20	24	14	18	15	15	16	14	
3	188	97	69	108	53	30	67	28	23	38	20	19	23	15	26	21	18	16	14	15	13	12	22*	18	

	2.4			3.0			3.6			4.0			4.4			4.6			5.0			
	Temperature (°F)																					
Replication	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	35	59	72	45	75	78	48	89	94	52	81	86*	57	79*	110	61	107*	94	59	106	124*	1"
2	38	57	70	46	73	84	43	77*	82*	61	81	101	65	90	85*	71	96	106	61	100	114	
3	32	65*	75	50	73	79	48	86	92	56	80	102	68	95	104	70	95	100	69	89*	109	
1	16	25	30	12	22	32*	18	28	31	15	29	35	12	24*	36	21	37	39	23	37	45	2"
2	16	24	25	12	23	25	15	29	31	19	30	33	15	34	41	23	36	41	26	39	40	
3	18	24	27	18	25	28	16	27	35	18	27	37	16	31	39	24	36	41	24	40	43	
1	18	14	16	12	15	17	10	18	18	9	15	17	12	15	18	10	20	23	13	20	49*	3"
2	12	13	15	10	14	17	9	16	12	9	13	20	12	9*	15	11	15	18	9	17	21	
3	16	16	17	8	11	29	10	15	17	7	14	18	9	18	20	10	19	20	11	16	25	

MINICEL - 18 in. Drop Height

STRESS LEVELS (PSI)

Thickness	Replication	0.04			0.10			0.20			0.40			0.80			1.0			1.6			2.0		
		Temperatures (°F)																							
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	248	187	113	182	91	77	107	66	65	62	57	70	44	59	81	46	67	84	56	93	107	59	100	115
	2	252	200	120	179	95	72	113	61	66	57	61	66	45	66	79	42	72	85	58	95	117*	68	96	118
	3	250	200	119	183	90	74	103	63	63	56	54	58	45	69	78	45	67	79	54	91	100	68	99	120
2"	1	259	166	86	166	65	52	88	48	37	50	39	35	31	33	35	26	30	56*	25	36	43	28	42	44
	2	250*	216*	105*	166	65	51	87	42	39	46	33	36	31	35	43	27	30	35	24	36	40	26	37	45
	3	260	128	91	157	67	53	73*	43	37	41*	35	40	33	32	37	27	30	36	26	35	44	28	38	48
3"	1	169	112	67	118*	56	41	92*	38	38*	48	29	28	28	29*	27	15	14	24	22	18	23	19	22	24
	2	198	142*	73	153	54	43	76	34	31	44	26	25	27	22	24	22	17	19	20	18	23	22	19	28
	3	242*	125	71	148	49	42	66	40	32	41	27	25	32	20	27	24	21	19	27	21	24	18	23	26

Thickness	Replication	2.4			3.0			3.6			4.0		
		Temperatures (°F)											
		-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	64*	128	134	100	146	174	104	175	164	131	187	210
	2	78	107*	133	99	143	163	124*	161	173	124	166*	211
	3	73	126	139	98	137	165	99	168	206*	100*	189	202
2"	1	27	44	48	32	52	63	38	64	68	41	73	78
	2	26	45	54	32	57	66	39	59	70	38	61*	84
	3	29	34*	54	31	57	59	37	61	75	35	68	72
3"	1	15	22	19*	18	24	32	18	29	38	25	38	39
	2	12	20	31	15	29	33	15	29	33	21	38	38
	3	15	21	28	12	27	34	19	30	35	21	34	37

MINICEL - 24 in. Drop Height

STRESS LEVELS (PSI)

	0.04			0.08			0.10			0.20			0.40			0.60			0.80			1.0			1.4			
	Temperature (°F)																											
Replication	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	231*	180	114	262	122	87	200	106	90	104	77	80	66	71	89	60	83	102	66	96	109	67	98	122	81	123	147	1"
2	335	215*	121	263	127	104	195	107	87	104	73	81	63	69	86	61	82	97	60	102	108	60	101	122	79	123	150	
3	289	195	113	250*	116	104	198	106	86	97	77	85	63	68	82	61	73*	103	57	88*	120	61	86*	111	62*	124	150	
1	277*	163	85	235	112	69	194*	91*	61	102	55	44	48	39	43	49	37	46	35	36	45	34	38	45	31	46	54	2"
2	254	202*	88	125*	101*	77	213	82	60	88	49	43	50	39	36	41	38	42	36	34	45	35	37	47	28	43	52	
3	245	149	97	270	114	75	205	81	57	93	51	46	71*	41	42	40	34	40	35	38	41	31	36	49	29	39	51	
1	220*	111*	94	233	104*	53	110*	68	50	73*	39	39	53	36	27	36	24	25	27	27	24	30	24	27	24	23	27	3"
2	262	167	86	192*	81	53	155	75*	45	87	40	32	47	33	26	37	23	27	29	25	26	24	32*	25	22	27	30	
3	281	158	68*	232	82	57	169	64	44	89	42	34	46	27*	27	37	25	27	31	24	26	30	23	29	21	24	27	

	1.8			2.0			2.2			2.4			2.6			3.0			3.4			4.0			
	Temperature (°F)																								
Replication	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	112	176*	186	96	148	208				158*	198	235													1"
2	108	139	166*	92	164*	205				145	203	254													
3	90*	137	192	98	136	190				142	198	246													
1	30	49	60	32	16*	64	41	81*	86	47	81	95	52	86	95	45	83	93	57	108	114	68	107	116	2"
2	32	50	65	32	54	66	46	76	92	44	73	90	48	88	99	45	84	92	58	103	107	75	106	116	
3	33	49	60	30	43	62	42	76	98	49	70	100	102*	87	100	57	78	101	44	99	102	59	100	115	
1	21	25	30	23	26	34	24	34	45	22	26	45	21	42	47	21	40	49	22	46	51	30	55	63	3"
2	21	27	35	22	23	32	28	34	48	18	28	49	24	46	49	20	36	51	20	43	54	31	58	65	
3	20	27	31	19	25	33	23	38	50	24	25	51	27	44	51	31*	41	43	22	43	49	26	56	52*	

MINICELL - 30 in. Drop Height

STRESS LEVELS (PSI)

Temperature (°F)

Thickness	Replication	0.04			0.08			0.1			0.2			0.3			0.4			0.6		
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	365*	208*	143	264	133	118	240	115	100	110	91	98	84	95	102	76	90	120	75	98	144
	2	334	195	136	271	129	113	220	100	112	112	88	96	84	93	101	85	91	116	72	106	127
	3	321	187	137	263	134	109	210	100	105	116	92	99	90	93	98	78	90	120	67	103	138
2"	1	372	173	91	258	99	76	244*	95	65	98	64	54	84	46	49	54	46	53	48	49	54
	2	369	169	96	241*	104	76	220	80*	72	106	56	55	82	49	52	57	46	47	47	45	50
	3	369	192*	106*	269	56*	82	215	88	72	88*	58	53	91	54	49	57	49	53	46	42	55
3"	1	330*	177*	106*	278*	90	60	184	64	56	77	52*	41	71	36	33	50	32	31	38	30	31
	2	295	166	89	265	90	57	181	70	58	79	45	39	60*	37	35	52	36	32	43	29	34
	3	301	171	81	257	84	58	195*	80*	57	87*	43	41	76	33	33	46*	35	33	46	30	34

Temperature (°F)

Thickness	Replication	0.7			0.8			1.0			1.2			1.3			1.4			1.6		
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	71	108	150	66	113*	170*	79	136	182	105	181	212	99	203*	242	108	201	222	140	217*	287*
	2	70	119	147	76	133	145	86	170*	161*	88	173	213	102	178	229	107	206	263*	111	234	264
	3	69	114	145	77	130	154	78	149	191	88	139*	211	97	182	228	125	204	233	123	232	256
2"	1	41	46	53	41	48	61	38	53	63	37	51	67	34	57	68	32	60	83	35	73	89
	2	42	46	56	39	43	58	40	54	62	36	52	72	35	54	73	39	55	77	41	70	41*
	3	47	47	58	40	48	55	40	57	65	37	56	69	35	63	69	40	62	79	37	66	81
3"	1	37	30	29	36	31	33	35	32	36	27	33	36	25	34	35	27	32	37	23	35	44
	2	38	28	32	30*	32	33	31	29	34	27	31	35	24	32	39	25	29	39	24	33	43
	3	36	29	31	35	30	34	32	32	34	26	28	35	27	32	37	26	32	36	26	31	43

MINICEL - 30 in. Drop Height
(Continued)

STRESS LEVELS (PSI)

Replication	1.8			2.0			2.2			2.4			2.6			3.0			3.4			
	Temperature (°F)																					
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	140	230	318	184	254	335																1"
2	129	207	315	147	252	334																
3	142	262*	286*	147	229*	322																
1	39	78	90*	44	75	106	56	102	108	59	93	119	51	111	131	76	99*	139	79	135	162	2"
2	36	77	101	49	77	108	54	99	101	51	92	122	59	105	121*	68	128	141	75	125	159	
3	41	72	108	37*	75	105	54	96	110	56	107*	125	51	103	130	68	116	137	89*	105*	157	
1	26	41*	43	25	39	47	22	45	51	30	49	59	28	50	63	26	51	66	32	65	74	3"
2	27	33	45	22	41	51*	23	48	50	30	44	55*	27	29*	56*	37*	54	66	32	64	72	
3	25	34	42	23	34	46	26	46	52	26	44	62	26	48	60	29	48	63	34	50*	60*	

*These values were removed by the outlier procedure during the UAH analysis.

CORRELATION COEFFICIENTS OF THE BEST FITTING POLYNOMIALS IN THE UAH STUDY

Temperature (°F)	Thickness (in.)	Drop Height 12 in.	18 in.	24 in.	30 in.
-65	1	0.97	0.98	0.98	0.97
	2	0.99	0.99	0.96	0.99
	3	0.99	0.98	0.99	0.98
70	1	0.98	0.98	0.99	0.98
	2	0.98	0.96	0.97	0.98
	3	0.97	0.98	0.97	0.97
160	1	0.98	0.98	0.97	0.98
	2	0.97	0.97	0.94	0.97
	3	0.93	0.98	0.96	0.97

NOTE: This table gives the sample correlation coefficients from an analysis of regression variance between the data and a polynomial fit of the form

$$y_i = b_0 + b_1 \ell n x_i + b_2 (\ell n x_i)^2.$$

SELECTED BEST FITTING POLYNOMIALS IN THE UAH STUDY

(Hercules Minicel, 2 lb/ft^3 Density)

Thickness (in.)	Temperature (°F)	Design Curve Equation
	12 in. Drop Height	
1	-65	$y = 377.74 - 142.48 \ln x + 14.78 (\ln x)^2$
	70	$y = 278.24 - 118.34 \ln x + 14.44 (\ln x)^2$
	160	$y = 197.11 - 84.21 \ln x + 11.27 (\ln x)^2$
2	-65	$y = 367.89 - 131.51 \ln x + 12.22 (\ln x)^2$
	70	$y = 201.97 - 78.34 \ln x + 8.34 (\ln x)^2$
	160	$y = 142.03 - 55.43 \ln x + 6.31 (\ln x)^2$
3	-65	$y = 329.67 - 118.13 \ln x + 10.86 (\ln x)^2$
	70	$y = 159.93 - 58.41 \ln x + 5.77 (\ln x)^2$
	160	$y = 105.43 - 36.66 \ln x + 3.73 (\ln x)^2$
	24 in. Drop Height	
1	-65	$y = 691.02 - 301.76 \ln x + 36.16 (\ln x)^2$
	70	$y = 403.76 - 193.48 \ln x + 27.86 (\ln x)^2$
	160	$y = 280.51 - 141.90 \ln x + 23.96 (\ln x)^2$
2	-65	$y = 544.94 - 210.69 \ln x + 21.69 (\ln x)^2$
	70	$y = 333.50 - 150.03 \ln x + 18.71 (\ln x)^2$
	160	$y = 202.97 - 93.58 \ln x + 13.17 (\ln x)^2$
3	-65	$y = 517.16 - 194.42 \ln x + 18.94 (\ln x)^2$
	70	$y = 289.25 - 123.01 \ln x + 13.98 (\ln x)^2$
	160	$y = 170.46 - 73.58 \ln x + 9.21 (\ln x)^2$

Appendix B

STEPWISE REGRESSION PROGRAM LISTING

```
C STEPWISE MULTIPLE LINEAR REGRESSION
C
C         WRITTEN BY WAYNE L. JONES,  REDSTONE ARSENAL, ALABAMA
C         BASED UPON PROCEDURES IN DRAPER"S APPLIED REGRESSION ANALYSIS
C         AND SHARE NUMBER 1333
C         TAPES 11  AND 10  ARE USED AS BINARY INPUT TAPES.
C         TAPES  9  AND 10  ARE USED AS WORK TAPES.
C**********                                                           **********
C A. MLR CONTROL CARD 1  FORMAT( 14I5 ) ****************************************
C 01-05 NPROB  = NUMBER TO IDENTIFY PROBLEM.                                   *
C 06-10 NXV    = TOTAL NUMBER OF INDEPENDENT VARIABLES IN INPUT DATA.          *
C 11-15 NYV    = TOTAL NUMBER OF   DEPENDENT VARIABLES IN INPUT DATA.          *
C 16-20 INDEXY = INDEX OF THE DEPENDENT VARIABLE FOR THE PROBLEM.              *
C 21-25 NDATA  = TOTAL NUMBER OF DATA OBSERVATIONS FOR THE PROBLEM.            *
C                   IF UNKNOWN- SET EQUAL MAXIMUM EXPECTED AND SET LAST        *
C                   DATA OBSERVATION EQUAL TO  99999999.                       *
C 26-30 IDEN   = NUMBER OF ALPHABETIC HEADER CARDS ( SEE C ).                  *
C 31-35 INTYPE = 0 FOR REGULAR RUN WITH DATA ON CARDS.                         *
C                1 TO REWIND 10 AND STORE CARD DATA FOR LATER PROBLEM.         *
C                2 TO REWIND 10 AND USE DATA STORED BY A PREVIOUS PROB.        *
C                3 TO STORE CARD DATA ON TAPE 10 WITHOUT REWIND.               *
C                4 TO USE DATA ON TAPE 10 WITHOUT FIRST REWINDING.             *
C                5 TO USE TAPE 11 AS INPUT AFTER REWINDING.                    *
C                6 TO USE TAPE 11 AS INPUT WITHOUT REWINDING.                  *
C                7 REWIND 11, USE AS INPUT, THEN REWIND FOR LATER USE.         *
C 36-40 NREAP  = 0 TO USE DATA WITHOUT REARRANGING IT.                         *
C                1 TO REARRANGE DATA ACCORDING TO CONTROL CARD F.              *
C 41-45 MAXSTP = MAXIMUM NUMBER OF STEPS OR ITERATIONS ALLOWED.                *
C                  TO BYPASS PRINTOUT OF CALCULATIONS PRIOR TO SUMMARY,        *
C                  SET EQUAL TO 999.                                           *
C 46-50 IFBACK = STEP AT WHICH BACK SOLUTION STARTS (ACTUAL VS PRED.).         *
C                  SET EQUAL TO 0 FOR NO BACK SOLUTION.                        *
C                  SET EQUAL TO 999 FOR BACK SOLUTION OF SUMMARY ONLY.         *
C                  NOTE - IF NDATA*(NXV+1) IS GREATER THAN 3000, TAPE 9        *
C                  IS USED TO STORE DATA THEREBY INCREASING RUN TIME.          *
C 51-55 NSTART = NUMBER OF INDEPENDENT VARIABLES THAT YOU WISH TO START        *
C                  THE REGRESSION WITH (SEE D).    NORMAL VALUE IS 0.          *
C                  IF NSTART = -1 THE PROGRAM WILL AUTOMATICALLY PUT           *
C                  ALL NXV VARIABLES IN REGRESSION AT START WITH A             *
C                  TEST OF ONE, WITHOUT CONTROL CARDS IN D. TEST IS ZERO       *
C                  FOR OTHER NEGATIVE VALUES.                                  *
C 56-60 MINSUM = MIN NBR OF IND VAR IN SUMMARY O/P. NORMAL VALUE IS 1.         *
C 61-65 MAXSUM = MAX NBR OF IND VAR IN SUMMARY O/P. NORMAL VALUE IS NXV.       *
C 66-70 MAXREG = MAX NBR OF IND VAR IN REGRESSION.  NORMAL VALUE IS NXV.       *
C**********                                                           **********
C B. MLP CONTROL CARD 2  ***************                                       *
C 01-05 IFWT   = 0 FOR UNWEIGHTED DATA                                         *
C              = 1 IF WEIGHTS ARE READ IN AS INPUT.                            *
C 06-10 TFCNST = 0 IF CONSTANT TERM IS TO BE CALCULATED.                       *
C              = 1 TO DELETE CONSTANT TERM                                     *
C              =-1 IF CONST TERM IS TO BE CONSIDERED AS THE COEFFICIENT        *
C                  OF A NEW INDEPENDENT VARIABLE X0 WHICH ALWAYS HAS THE       *
C                  VALUE 1. THE SIGNIFICANCE OF THE CONSTANT WILL BE           *
C                  INDICATED BY ITS STANDARD ERROR.                            *
C 11-15 IFLIST = 0 TO LIST INPUT DATA, 1 OTHERWISE.                            *
C 16-20 IFSUMS = 0 TO LIST SUM(XI*XJ), 1 OTHERWISE.                            *
C 21-25 IFRES  = 0 TO LIST SUM(XI-XBARI)(XJ-XBARJ), 1 OTHERWISE.               *
```

```
C 26-30 IFCORR = 0 TO LIST SIMPLE CORRELATION COEFFICIENTS, 1 OTHERWISE.     *
C 31-35 NCROSS = 1 IF YOU WISH TO INCLUDE AS ADDITIONAL INDEPENDENT          *
C                  VARIABLES. THE SQUARES AND THE CROSS PRODUCTS OF          *
C                  THE INDEPENDENT VARIABLES.  ZERO OTHERWISE.               *
C                  GENERATED VARIBLES ARE X(NXV+1)=X1*X1                     *
C                  X(NXV+2)=X1*X2, X(NXV+3)=X1*X3,...X(NXV+NXV)=X1*XNXV      *
C                  X(NXV+NXV+1)=X2*X2, X(NXV+NXV+2)=X2*X3 .... ETC.          *
C              = N (GREATER THAN 1) FOR POLYNOMIAL CURVE FIT OF ORDER N.     *
C                NOTE - NSTART SHOULD BE SET TO -2 FOR NORMAL RUN.           *
C 36-40 IFTRA  = ALLOWS FOR TRANSFORMATIONS OF INPUT DATA. SEE G FOR USE.    *
C              = 0 FOR NO TRANSFORMATIONS                                    *
C              = 1 FOR TRANSFORMATIONS.                                      *
C              =-1 TO USE PREVIOUS TRANS WHICH ARE STILL IN CORE.            *
C 41-45 NVOID  = 0 TO PROCESS ALL OBSERVATIONS. = 1 TO READ UP TO 14         *
C                OBSERVATIONS TO BE LEFT OUT OF REGRESSION. SEE CONTROL      *
C                CARD E. = -1 TO USE PREVIOUS E CARD .                       *
C 46-50 IFSUB  = ZERO FOR NORMAL RUN. POSITIVE VALUE CALLS IN A              *
C                USER SUPPLIED SUBROUTINE CALLED  NEROFX(IFSUB,NXV)          *
C                TO CHANGE NXV (NOTE- NDROFX MUST BE SUPPLIED EVEN           *
C                IF IT IS JUST A RETURN). A USER SUPPLIED SUBROUTINE         *
C                CALLED  EQUAT(IFSUB,DATA) IS USED TO MAKE THE DESIRED       *
C                CROSS PRODUCT AND TRANSFORMATIONS (NOTE- DEPENDENT          *
C                VARIABLE SHOULD BE DEFINED AS THE VARIABLE DATA(NXV+1)      *
C                WHERE DATA IS A SET OF OBSERVATIONS GOING INTO THE          *
C                S/R AND THE TRANSFORMED SET COMMING OUT ).                  *
C 51-55 NFMT   = 0 FOR REGULAR INPUT FORMAT(7F10.0) .                        *
C              = 1 TO READ INPUT FORMAT( SEE I ).                            *
C              =-1 TO USE FORMAT FROM PREVIOUS RUN.                          *
C 56-60 IFPNCH = 0 TO DELETE PUNCHING OF EQUATION COEFFICIENTS IN SUMMARY.   *
C 61-65 IFDATE = 0 TO PRINT DATE OF COMPUTER RUN                             *
C 66-70 IFNAME = 0 TO READ NAMES OF VARIABLES. SEE H FOR FORMAT              *
C              =-1 TO USE PREVIOUS H CARD, STILL IN CORE.                    *
C              = 1 TO ASSUME BLANK NAMES                                     *
C**********                                                         *********
C C. ALPHABETIC HEADER  CARDS.     DO NOT USE IF  IDEN=0 .                   *
C     IDEN CARDS WITH FORMAT(16A5)       LAST CARD REPEATED ON EACH PAGE     *
C**********                                                         *********
C D. CARDS FOR VARIABLES IN REGRESSION AT START AND CORRESPONDING TESTS.     *
C     DO NOT USE IF NSTART=0. THERE SHOULD BE "NSTART" FIELDS.7(F8.0,I2)     *
C 01-08 TEST(1) = A TEST CONDITION WHICH DETERMINES WHETHER A VARIABLE      *
C                  WILL BE DELETED OR ADDED TO THE REGRESSION. ITS           *
C                  VALUE IS 1-R**2. ZERO CORRESPONDS TO A MULTIPLE CORR      *
C                  COEFFICIENT OF 1, WHICH MAKES IT IMPOSSIBLE FOR THE       *
C                  PROGRAM TO DELETE THAT VARIABLE FROM THE SET OF IND       *
C                  VARIABLES. TEST=1 CORRESPONDS TO MULT CORR COEF OF 0,     *
C                  WHICH MAKES SUCH A DELETION CERTAIN.                      *
C 09-10 INDFX(1)= FIRST  VARIABLE TO BE INCLUDED IN REGRESSION AT START.     *
C 11-18 TEST (2)= TEST FOR TWO VARIABLE SET.                                 *
C 19-20 INDFX(2)= SECOND VARIABLE TO BE INCLUDED IN REGRESSION AT START.     *
C 21-28 TEST (3)= TEST FOR THREE VARIABLE SET.                               *
C 29-30 INDFX(3)= ETC.                                                       *
C**********                                                         *********
C E. OBSERVATIONS TO BE REMOVED FROM REGRESSION. USE ONLY IF  NVOID=1 .      *
C 01-05 NOGOOD(1) = INDEX OF 1ST POINT TO BE REMOVED. ( FORMAT(14I5) )       *
C 06-10 NOGOOD(2) = ETC.                                                     *
C                                                                            *
C**********                                                         *********
```

```
C F. CONTROL CARO TO REARRANGE MLR DATA.  DO NOT USE IF NREAR=0.  (14I5)      *
C 01-05 NWCRDS = NUMBER OF WORDS IN TAPE OR CARD RECORD.                       *
C 06-10 LOCY   = LOCATION OF DEPENDENT VARIABLE Y.                             *
C 10-15 LOCX(J),J=1,NXV = LOCATIONS OF INDEPENDENT VARIABLES.                  *
C                  IF  IFWT=0 LAST LOCATION IS FOR WEIGHTS.                    *
C**********                                                           **********
C G. TRANSFORMATION CONTROLS. DO NOT USE IF IFTRA=0.  FORMAT(7(F8.0,I2))       *
C     PUT TRANSFORMATIONS AND CORRESPONDING CONSTANTS IN SAME ORDER AS         *
C     X AND Y VARIABLES.                                                       *
C     TRANSFORMATIONS  0=NONE, 1=X+C, 2=X*C, 3=X/C, 4=C/X, 5=X**C,             *
C     6=C**X, 7=LN(X+C), 8=LOG(X+C), 9=E**(C*X), 10=E**(C/X),                  *
C     11=SIN(C*X), 12=COS(C*X), 13=TAN(C*X)                                    *
C 01-08 CONST  = CONSTANT FOR FIRST VARIABLE                                   *
C 09-10 NBRTRA = TRANSFORMATION FOR FIRST VARIABLE                             *
C 11-18        = CONST FOR SECOND VARIABLE ,  ETC.                             *
C 19-20        = TRA FOR SECOND VARIABLE,     ETC.                             *
C**********                                                           **********
C H. INPUT VARIABLE NAMES IN ORDER OF INPUT.  USE ONLY IF IFNAME=C.            *
C 01-10 = NAME OF FIRST INPUT VARIABLE                                         *
C 11-20 = NAME OF SECOND INPUT VARIABLE, ETC.     FORMAT( 7(A6,A4) )           *
C**********                                                           **********
C I. VARIABLE FORMAT FOR INPUT DATA (12A6). USE ONLY IF NFMT=1.                *
C**********                                                           **********
C J. MLR DATA CARDS SHOULD BE PUNCHED WITH FORMAT(7F10.0), OBSERVATION         *
C     BY OBSERVATION IN THE FOLLOWING ORDER (IF INTYPE=0,1) X1, X2, X3,        *
C     X4, ..., XNXV,  Y1, Y2, Y3, ..., YNYV,  WT(IF IFWT=1 ).                  *
C    DATA CARDS FOR INTYPE=0,1,3 ONLY. BINARY TAPE INPUT FOR OTHER CODES.      *
C    IF NREAR=1 ORDER OF DATA IS DETERMINED BY CONTROL CARD F.                 *
C    OBSERVATIONS WITH BLANK DATA ( -0 ) ARE REMOVED FROM REGRESSION.          *
C**********                                                           **********
C                                                                              *
C TO USE THE PROGRAM FOR AN ORDINARY MULTIPLE REGRESSION (I.E. NO              *
C  ADJOINING OR DELETING), PUT ALL VARIABLES IN THE REGRESSION AT              *
C  THE OUTSET ( NSTART = -NXV ) AND PUT MAXSTP = 1.                            *
C*******************************************************************************

C********* CARD OUTPUT  ( IF IFPNCH IS NOT EQUAL ZERO )
C
C          ONE CARD FOR EACH VARIABLE IN EQUATION
C          FORMAT( I5,E20.8,5I5,F20.4,F10.7 )
C     01-05 = I = INDEX OF INDEPENDENT VARIABLES IN EQUATION
C     06-25 = COEFF(I) = COEFFICIENT FOR VARIABLE I
C     26-30 = NPROB  = PROBLEM NUMBER
C     31-35 = NBRNOW = NUMBER OF VARIABLES IN EQUATION
C     36-40 = INDEXY = INDEX OF DEPENDENT VARIABLE
C     41-45 = NOSTEP = STEP NUMBER IN WHICH THE EQUATION WAS COMPLETED
C     46-50 = IFPNCH = INPUT VALUE GREATER THAN ZERO                      MLR 1550
C     51-70 = SIGPCT = STANDARD ERROR OF EQUATION AS A PERCENT OF Y MEAN
C     71-80 = REGRCO = CORRELATION COEFFICIENT OF EQUATION
C
C********* BASIC STATISTICS OUTPUT **********
C
C          XI = X = VALUE OF OBSERVATION FOR VARIABLE I
C          SUM( XI ) = SUMMATION OF VARIABLE I
C          N = NUMBER OF OBSERVATIONS
C          XN = WEIGHTED NUMBER OF OBSERVATIONS
C
```

```
C      MEAN = WEIGHTED AVERAGE = SUM( XI ) / XN
C      STANDARD DEVIATION = SQRT( (SUM(XI**2)-SUM(XI)*MEAN)/(XN-1) )
C
C      SUM OF VARIABLES  = SUM( XI )
C      RAW SUM OF SQUARES AND CROSS PRODUCTS = SUM( XI*XJ )
C      SUM OF SQUARES AND CROSS PRODUCTS ABOUT THE MEAN = CORRECTED SUMS=
C           = SS( I,J ) = SUM(XI*XJ)-SUM(XI)*SUM(XJ)/XN
C      SIMPLE CORRELATION COEFFICIENTS =
C           = R(I,J) = SS(I,J)/SQRT( SS(I,I)*SS(J,J) )
C
C********* RESIDUAL ANALYSIS ( ACTUAL VS PREDICTED ) PRINTOUT *********
C
C      ACTUAL = Y = DEPENDENT VARIABLE
C      PREDICTED = YC = COMPUTED Y USING REGRESSION EQUATION
C                     = A0 + A1*X1 + A2*X2 + ... + AN*XN
C      RESIDUAL = E = YC - Y
C      NORMALIZED DEVIATE = RESIDUAL / STANDARD ERROR
C      PERCENT DEVIATION  = 100 * RESIDUAL / ACTUAL
C      WEIGHT = INPUT WEIGHT OF OBSERVATION
C      SSE = RESIDUAL SUM OF SQUARES
C      CHI SQUARE = SUM( ( RESIDUAL**2 ) / YC )
C
C********* ANALYSIS PRINTOUT *********
C
C      TOTAL(CORRECTED) SUM OF SQUARES = SUM OF SQUARES ABOUT THE MEAN
C          = SUM((Y-YMEAN)**2) = SUM((YC-YMEAN)**2)+SUM((Y-YC)**2) =SS(Y)
C      TOTAL( ORIGIN ) SUM OF SQUARES = SUM OF SQUARES ABOUT THE ORIGIN.
C          USED INSTEAD OF SS(Y) WHEN REGRESSION IS FORCED THRU ORIGIN.
C      REGRESSION SUM OF SQUARES = SUM OF SQUARES DUE TO REGRESSION
C          = EXPLAINED VARIATION = SUM((YC-YMEAN)**2) = SS(R)
C      RESIDUAL SUM OF SQUARES =   SUM OF SQUARES ABOUT THE REGRESSION
C          = UNEXPLAINED VARIATION = SUM((Y-YC)**2) = SS(E)
C      THE MEAN SQUARES COLUMN IS OBTAINED BY DIVIDING THE SUM OF SQUARES
C          ENTRY BY ITS CORRESPONDING DEGREES OF FREEDOM.
C      RESIDUAL MEAN SQUARE = VARIANCE ABOUT THE REGRESSION = S**2 =MS(E)
C      COEFFICIENT OF MULTIPLE DETERMINATION = PCT OF EXPLAINED VARIATION
C          = (SS DUE TO REGRESSION)/(SS ABOUT MEAN) = CORR. FORM SS(R)
C      CORRELATION COEFFICIENT = R = SQRT( CORR. FORM OF SS(R) )
C      STANDARD ERROR OF ESTIMATE = S = SQRT( MS(E) )
C      S.E. AS PCT. OF MEAN = 100 * S / YMEAN
C      F  TEST FOR SIGNIFICANCE OF REGRESSION = MS(R)/MS(E)
C*
C      CONSTANT = A(0) = YMEAN-SUM(A(I)*XMEAN(I)) = CONSTANT TERM
C      COEFFICIENT = A(I) = THE EFFECT ON Y OF A UNIT INCREASE IN XI WHEN
C          THE OTHER VARIABLES ARE HELD CONSTANT.
C      STANDARD ERROR = STANDARD ERROR OF REGRESSION COEFFICIENT.
C          THE 95 PERCENT CONFIDENCE LIMITS FOR A UNIVERSE REGRESSION
C          COEFFICIENT ARE GIVEN BY THE SAMPLE COEFFICIENT PLUS AND MINUS
C          T(0.025) TIMES THE ESTIMATED STANDARD OF THE COEFFICIENT.
C      COEFF/SE = USED IN HYPOTHESIS THAT COEFFICIENT = 0.
C          = COEFFICIENT DIVIDED BY ITS STANDARD ERROR TO GIVE THE NUMBER
C          OF S.E. AWAY FROM HYPOTHESIZED ZERO. SHOULD BE GREATER THAN T
C          VALUE TO REJECT THE HYPOTHESIS THAT THE COEFFICIENT IS NOT
C          SIGNIFICANTLY DIFFERENT FROM ZERO.
C      F   = F VALUE TO REMOVE VARIABLE FROM REGRESSION. **** NOT USED **
C      BETA COEFFICIENT = MEASURE OF THE NET EFFECT OF EACH VARIABLE ON Y.
C      RSQ CHANGE = DECREASE IN RSQ IF THE VARIABLE IS REMOVED FROM REGRESSION
```

```
C*
C     PARTIAL RSQ = THE SQUARE OF THE PARTIAL CORRELATION COEFFICIENT OF
C         VARIABLE K NOT IN THE REGRESSION WITH THE RESPONSE Y.
C         = R(KY.LMN... )**2 WHERE L,M,N,... ARE ALREADY IN REGRESSION.
C         = RELATIVE AMOUNT OF IMPROVEMENT THAT IS BROUGHT ABOUT IF
C         VARIABLE K WERE ADDED TO THE REGRESSION.
C     NORMED SUM/SD = THE NORMALIZED SUM OF SQUARES OF RESIDUALS FOR
C         VARIABLE K  HAD IT TOO BEEN REGRESSED. USEFUL IN DRAWING
C         ATTENTION TO NEAR-LINEAR DEPENDENCIES AMONG THE IND. VARIABLES
C     DELTA RSQ = CHANGE IN RSQ IF VARIABLE K WERE ADDED TO REGRESSION.
C         VARIABLE WITH LARGEST DELTA IS ADDED TO REGRESSION NEXT.
C     F    = F VALUE TO ADD VARIABLE TO REGRESSION. **** NOT USED ****
C
C********* ADDING  AND  DELETING  VARIABLES ***************************
C
C STEP 1 - THE VARIABLE NOT IN THE EQUATION WHICH CAUSES THE GREATEST
C     CHANGE IN RSQ IS ADDED TO THE REGRESSION.
C STEP 2 - THE VARIABLES IN THE EQUATION ARE THEN CHECKED TO SEE IF ONE
C     CAN BE DELETED. THE VARIABLE WHICH CAUSES THE SMALLEST CHANGE IN
C     RSQ IS SELECTED FOR REMOVAL. IF THE EQUATION WITHOUT THIS VARIABLE
C     PRODUCES A SS(R) WHICH IS SMALLER THAN THE PREVIOUS SS(R) FOR THAT
C     NUMBER OF VARIABLES, THE VARIABLE IS REMOVED.
C STEP 3 - IF A VARIABLE WAS REMOVED, REPEAT STEP 2.
C          OTHERWISE REPEAT STEP 1 AND 2.
C
C*********************************************************************
C
C     IT SHOULD BE NOTED THAT THE STATISTICS FOR NON-LINEAR EQUATIONS
C     SHOULD BE USED WITH CARE, AND SHOULD NOT BE COMPARED WITH THOSE
C     FROM LINEAR EQUATIONS, AS THEY HAVE DIFFERENT MEANINGS.
C     FOR EXAMPLE - IF Y IS TRANSFORMED BY TAKING ITS LOGARITHM, THE
C     SUM OF THE SQUARES OF THE ACTUAL RESIDUALS BETWEEN THE CALCULATED
C     AND THE OBSERVED Y VALUES ARE NOT  MINIMIZED, RATHER THE SUM OF
C     SQUARES OF THE LOGARITHMS OF THE  RATIOS  OF THESE VALUES ARE
C     BEING MINIMIZED ( (LOG YC - LOG Y) = ( LOG(YC/Y) ) ).
C     THEREFORE, COMPARISION OF ANY STATISTICS THAT ARE BASED UPON THE
C     SUM OF THE SQUARES OF THE Y RESIDUALS SUCH AS THE F VALUE OR
C     CORRELATION COEFFICIENT MAY BE MISLEADING.
C
C     IT SHOULD ALSO BE NOTED THAT WHEN THE CURVE IS FORCED THROUGH THE
C     ORIGIN OR SOME OTHER SPECIFIED Y INTERCEPT, THE DEGREES OF FREEDOM
C     ARE CHANGED AND THE CURVE NO LONGER GOES THROUGH THE MEANS OF THE
C     VARIABLES, THEREBY, CHANGING THE VALUES OF THE STATISTICS AND
C     MAKING COMPARISIONS OF CURVES WITH UNSPECIFIED Y INTERCEPTS
C     MISLEADING.   ALSO, COMPARISON OF F VALUES WITH THE STANDARD F
C     DISTRIBUTION IS NOT NECESSARILY VALID.
C*********************************************************************
C
C     THE USERS OF THIS PROGRAM ARE URGED TO REVIEW THE STANDARD TEXTS
C     ON REGRESSION ANALYSIS FOR THE USES AND LIMITATIONS OF THIS
C     TECHNIQUE, AND BEAR IN MIND THAT THE STATISTICAL RELATIONSHIPS ARE
C     NO BETTER THAN THE DATA THAT WAS USED TO COMPUTE THEM.
C*******************************************************************

C
C
```

74/74   OPT=1   FTN 4.2+74278   11/22/74 10.10.21.

```
      PROGRAM  MLR
     * (INPUT=1010,OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT, PUNCH=2020,
     *      TAPE7=PUNCH, TAPE9=10019, TAPF10=10010, TAPE11=1001E )
C
C          WAYNE L. JONES
C
      COMMON SIGMA(60),A(52,52),SIMCOR(52,52),AVG(60),TEST(60)           MLR 2490
      COMMON POINT(60),STRING(3000),INDPAC(30,30),INCEXP(61)             MLR 2500
      COMMON INDEX(60),NCUT(60),KSTEP(60),ALPHA(16),YMEAN,IDEN,IFAVE     MLR 2510
      COMMON MAXSTP,IFPNCH,NSUMRY,NSKIP,NTAPE9,NEW                       MLR 2520
      COMMON NOVAR,NURNOW,NCSTEP,NDATA,NBRXYN,NERR,LFATH,DEFRM,NVAR      MLR 2530
      COMMON IFBACK,IFCNST,IFCORR,NPROB,NBRPVR,TCL,NCDEL,JF              MLR 2540
      COMMON INDEXY,LBAD,NOGOOD(28)                                      MLR 2550
      COMMON IFMT,YCONST,NYTRA,V(2,51),YTRA(2)                           MLR 2560
      COMMON  STDERR(50),CORSQR(50)                                      MLR 2570
      REAL               A,SIMCOR,SIGMA,AVG,TEST                         MLR 2580
C                                                                        MLR 2590
C                                                                        MLR 2600
      DIMENSION  VNAME(2,51),CROSS(8)                                    MLR 2610
      DIMENSION XDATA(255), LOCX(60), CONST(60), NBRTRA(60)
      DIMENSION  FMT(10)
      DIMENSION JVOID(14)                                                MLR 2640
      DIMENSION TRA(3), ATRA(2,17)                                       MLR 2650
      DATA  ATRA / 34*1H  /
      DATA (ATRA(1,J),J=1,17)/8HX + C   ,8HX * C   ,8HX / C   ,8HC / X   MLR 2660
     1 ,8HX**C    ,7HC**X    ,8HLN(X+C) ,8HLOG(X+C),8HE**(C*X),8HE**(C/X)MLR 2670
     2,8HSIN(C*X),8HCOS(C*X),8HTAN(C*X),9HSINH(C*D),9HCOSH(C*D),9HTANH(CMLR 2680
     3*D),11HCONST. TERM/                                                MLR 2690
      DATA (CROSS(J),J=1,8)/4HV(1),4HV(2),4HV(3),4HV(4),4HV(5),4HV(6),
     *                      4HV(7),4HV(8) /
      DATA  FMT / 8H(7E10.0) /
      DATA  ACT,UAL/6HACTUAL,1H /                                        MLR 4730
      DATA NREELS,NEOF/1,0/                                              MLR 2710
      DATA BLANK/1H /                                                    MLR 2730
      DATA AX/1HX/,AY/1HY/                                               MLR 2740
      DATA START,VOIDED,SEARCH,TRAN,FORAMT/6HSTART ,6HVOIDED,6HSEARCH,6HMLR 2750
     1 TRA= ,6HJFMT  /                                                   MLR 2760
C                                                                        MLR 2770
      DATA  XDATA(1)/ 60H1SELECTED PERCENTILE VALUES OF THE STUDENT T DIMLR 2780
     1STRIBUTION   /, XDATA(07)/78H              .10       .05       .01
     2   FOR    DIRECTIONAL (ONE-TAILED) TEST  /, XDATA(15)/78H   D.F.
     3   .20       .10       .02      FOR NONDIRECTIONAL (TWO-TAILED) TESMLR 2810
     4T   /                                                              MLR 2820
      DATA ( XDATA(J),J=37,188)/4H   1,3.078,6.313,31.821,4H   2,1.886,  MLR 2830
     1 2.920,6.965,4H   3,1.638,2.353,4.541,4H   4,1.533,2.132,3.747,    MLR 2840
     2 4H   5,1.476,2.015,3.365, 4H   6,1.440,1.943,3.143,               MLR 2850
     3 4H   7,1.415,1.895,2.998, 4H   8,1.397,1.860,2.896,               MLR 2860
     4 4H   9,1.383,1.833,2.821, 4H  10,1.372,1.812,2.764,               MLR 2870
     5 4H  11,1.363,1.796,2.718, 4H  12,1.356,1.782,2.681,               MLR 2880
     6 4H  13,1.350,1.771,2.650, 4H  14,1.345,1.761,2.624,               MLR 2890
     7 4H  15,1.341,1.753,2.602, 4H  16,1.337,1.746,2.538,               MLR 2900
     8 4H  17,1.333,1.740,2.567, 4H  18,1.330,1.734,2.552,               MLR 2910
```

```
      9 4H  19,1.328,1.729,2.539, 4H  20,1.325,1.725,2.520,            MLR 2920
      1 4H  21,1.323,1.721,2.518, 4H  22,1.321,1.717,2.508,            MLR 2930
      2 4H  23,1.319,1.714,2.500, 4H  24,1.318,1.711,2.492,            MLR 2940
      3 4H  25,1.316,1.708,2.485, 4H  26,1.315,1.706,2.479,            MLR 2950
      4 4H  27,1.314,1.703,2.473, 4H  28,1.313,1.701,2.467,            MLR 2960
      5 4H  29,1.311,1.699,2.462, 4H  30,1.310,1.697,2.457,            MLR 2970
      6 4H  40,1.303,1.684,2.423, 4H  50,1.298,1.676,2.403,            MLR 2980
      7 4H  60,1.296,1.671,2.390, 4H  80,1.292,1.664,2.374,            MLR 2990
      8 4H 100,1.290,1.660,2.365, 4H 200,1.286,1.653,2.345,            MLR 3000
      9 4H 500,1.283,1.648,2.334, 4H INF,1.282,1.645,2.326 /           MLR 3010
       WRITE(6,1080) ( XDATA(J), J=1,22 ), (XDATA(J),J=37,188)
C
       TSTART = SECOND(TSTART)
       PSTART = TSTART
C                                                                     MLR 3040
    10 CALL SLITE (0)                                                 MLR 3050
       DO 20 J=1,30                                                   MLR 3060
       DO 20 K=1,30                                                   MLR 3070
    20 INOPAC(J,K)=0                                                  MLR 3080
       JF = 0
C                                                                     MLR 3100
       READ (5,950) NPROB,NXV,NYV,INDEXY,NDATA,IDEN,INTYPE,NREAR,MAXSTP,IMLR 3110
      1FBACK,NSTART,MINSUM,MAXSUM,MAXREG,IFMT,IFCNST,IFLIST,IFSUMS,IFAVE,MLR 3120
      2IFCOPR,NCROSS,IFTRA,NSKIP,IFSUB,NFMT,IFPNCH,IFDATE,IFNAME       MLR 3130
       IF ( EOF(5) .NE. 0.0 )  GO TO 805
       WRITE (6,810)                                                  MLR 3090
C                                                                     MLR 3140
       WRITE (6,1010) NPROB,NXV,NYV,INDEXY,NDATA,IDEN,INTYPE,NREAR,MAXSTPMLR 3150
      1,IFBACK,NSTART,MINSUM,MAXSUM,MAXREG,IFMT,IFCNST,IFLIST,IFSUMS,IFAVMLR 3160
      2E,IFCOPR,NCROSS,IFTRA,NSKIP,IFSUB,NFMT,IFPNCH,IFDATE,IFNAME     MLR 3170
       D1=BLANK                                                       MLR 3180
C          DATE IS MAP S/R TO PICK DATE OFF SEQUENCE CARD             MLR 3220
       IF (IFDATE.EQ.0) CALL DATE (D1 )
       NBRNOW=NSTART                                                  MLR 3230
       IF( NFMT .EQ. 0 )  FMT(1) = RMT
       IF (NDATA.LE.0) NDATA=10000                                    MLR 3280
       NWTT=NXV+NYV                                                   MLR 3290
       IF (IFMT.NE.0) NWTT=NWTT+1                                     MLR 3300
       IF (IFSUB.GT.0) CALL NBROFX (IFSUB,NXV)                        MLR 3310
       NSTEP=0                                                        MLR 3320
       NSUMRY=0                                                       MLR 3330
       TOL=0.00001                                                    MLR 3340
       NBRXY=NXV+1                                                    MLR 3350
C          NBRX  = NUMBER OF INDEPENDENT VARIABLES                    MLR 3360
C          NBRXY = NUMBER OF INDEPENDENT VARIABLES + DEPENDENT VARIABLE  MLR 3370
C          NBRXYW= SIZE OF ARRAY = NBRXY + 1                          MLR 3380
C          NBRNOW = NUMBER OF COEFFICIENTS FOR PRESENT  EQUATION      MLR 3390
C          INDEX  = INDEX OF PRESENT  EQUATION                        MLR 3400
       IF (MAXSTP.EQ.0) MAXSTP=998                                    MLR 3410
       IF (MAXREG.LT.5) MAXREG=5                                      MLR 3420
C                                                                     MLR 3430
C        TO STORE DATA ON TAPE FOR USE IN ANOTHER PROBLEM, SET INTYPE=1  MLR 3440
C          FOR REWIND,  INTYPE=3 FOR NO REWIND                        MLR 3450
C          TO USE DATA FROM A PREVIOUS PROBLEM   SET INTYPE = 2 OR 5, MLR 3460
C          THEREBY CAUSING TAPE TO REWIND.                            MLR 3470
C          INTYPE = 4 OR 6 FOR BIN. TAPE 10 OR 11 TO BE USED AS INPUT MLR 3480
C                ALSO PREVENTS TAPE REWIND AT START OF PROBLEM.       MLR 3490
```

```
            NREWB5=1                                                                  MLR 3500
            IF (INTYPE.NE.7) GO TO 50                                                 MLR 3510
            NREWB5=0                                                                  MLR 3520
            INTYPE=5                                                                  MLR 3530
         50 NTAPE=10                                                                  MLR 3540
            NREAD=0                                                                   MLR 3550
            NWRITE=0                                                                  MLR 3560
            IF (INTYPE.EQ.0) GO TO 60                                                 MLR 3570
            IF (INTYPE.EQ.1.OR.INTYPE.EQ.3) NWRITE=1                                  MLR 3580
            IF (INTYPE.EQ.5.OR.INTYPE.EQ.6) NTAPE=11                                  MLR 3590
            IF (INTYPE.EQ.1.OR.INTYPE.EQ.2.OR.INTYPE.EQ.5) REWIND NTAPE               MLR 3600
            IF (INTYPE.NE.1.AND.INTYPE.NE.3) NREAD=1                                  MLR 3610
      C                                                                               MLR 3620
      C         NUMBER OF INDEPENDENT + DEPENDENT VARIABLES                           MLR 3630
         60 NTOTAL=NXV+NYV                                                            MLR 3640
            NEW=NXV+INDEXY                                                            MLR 3650
            IF (NTOTAL.LE.52) GO TO 70                                                MLR 3660
      C         TOO MANY VARIABLES                                                    MLR 3670
            NTOTAL=NEW                                                                MLR 3680
            IF (NTOTAL.LE.52) GO TO 70                                                MLR 3690
            WRITE (6,820) NTOTAL                                                      MLR 3700
            CALL EXIT                                                                 MLR 3710
         70 NOSTEP=0                                                                  MLR 3720
      C         CHECK FOR CROSS-PRODUCTS OR POLYNOMIAL
            IF (NCROSS.EQ.0) GO TO 80                                                 MLR 3730
            IF( NCROSS .GT. 1 ) GO TO 75
            WRITE(6,1110)
            NOVAR= (NBRXY*(NBRXY+1))/2                                                MLR 3740
            IF (NOVAR.LE.51) GO TO 90                                                 MLR 3750
            NCROSS=0                                                                  MLR 3760
            WRITE (6,050)                                                             MLR 3770
            GO TO 80
         75 IF( NCROSS .GT. 50 ) NCROSS = 50
            WRITE(6,1120) NCROSS
            IF( NXV .EQ. 1 ) GO TO 76
            NCROSS = 0
            WRITE(6,1130)
            GO TO 80
         76 NBRXY = NCROSS + 1
      C
      C
         80 NOVAR=NBRXY                                                               MLR 3780
         90 IF (IFCNST.LT.0) NOVAR=NOVAR+1                                            MLR 3790
            NBRXYW=NOVAR+1                                                            MLR 3800
            NBRX=NOVAR-1                                                              MLR 3810
            MAXVAR=MIN0(MAXSUM,MAXREG,NBRX)                                           MLR 3820
      C         READ CONTROL CARD C                                                   MLR 3830
            IF( IDEN ) 95, 94,110                                                     MLR 3840
         94 DO 100 J=1,16                                                             MLR 3850
        100 ALPHA(J)=BLANK                                                            MLR 3860
         95 IDEN = IABS(IDEN)                                                         MLR 3870
            GO TO 130                                                                 MLR 3880
        110 DO 120 I=1,IDEN                                                           MLR 3890
            READ (5,830) (ALPHA(J),J=1,16)                                            MLR 3900
        120 WRITE (6,840) (ALPHA(J),J=1,16)                                           MLR 3910
        130 CONTINUE                                                                  MLR 3920
      C         READ CONTROL CARD D                                                   MLR 3930
```

```
         DO 140 J=1,60                                               MLR 3940
         KSTEP(J)=1                                                  MLR 3950
     140 TEST(J)=+2.0E+30                                            MLR 3960
         IF (NSTART) 170,200,150                                     MLR 3970
     150 READ (5,930) (TEST(J),INDEX(J),J=1,NBRNCH)                  MLR 3980
C             PACK INDEX                                             MLR 3990
         DO 160 J=1,NBRNOW                                           MLR 4000
     160 CALL PACK (NBRNOW,J,INDEX(J),1)                             MLR 4010
         GO TO 190                                                   MLR 4020
CC170 NBRNOW=NBRX                                                    MLR 4030
     170 NBRNOW=NXV                                                  MLR 4040
         DO 180 J=1,NBRNOW                                           MLR 4050
         CALL PACK (NBRNOW,J,J,1)                                    MLR 4060
         INDEX(J) = J
         TEST(J) = 1.000                                             MLR 4070
     180 IF (NSTART.LT.-1) TEST(J)=0.0                               MLR 4080
     190 WRITE (6,940) START,(TEST(J),INDEX(J),J=1,NBRNCH)           MLR 4090
     200 MINVAR=MAX0(1,MINSUM)                                       MLR 4100
         NTAPE9=1                                                    MLR 4110
         IF (IFBACK.EQ.0) GO TO 210                                  MLR 4120
         IF (NDATA*NBRXYW.LE.3000) GO TO 210                         MLR 4130
         NTAPE9=0                                                    MLR 4140
         REWIND 9                                                    MLR 4150
     210 DO 220 I=1,NBRXYW                                           MLR 4160
         DO 220 J=1,NBRXYW                                           MLR 4170
     220 A(I,J)=0.0                                                  MLR 4180
C             READ CONTROL CARD E                                    MLR 4190
         IF (NSKIP.LE.0) GO TO 270                                   MLR 4200
         READ (5,950) (JVOID(J),J=1,14)                              MLR 4210
         JV=0                                                        MLR 4220
         DO 250 L=1,28,2                                             MLR 4230
         NOGOOD(L)=0                                                 MLR 4240
         NOGOOD(L+1)=0                                               MLR 4250
     230 IF (JV.EQ.14) GO TO 260                                     MLR 4260
         JV=JV+1                                                     MLR 4270
         IF (JVOID(JV)) 250,230,240                                  MLR 4280
     240 NOGOOD(L)=JVOID(JV)                                         MLR 4290
         NOGOOD(L+1)=JVOID(JV)                                       MLR 4300
         LBAD=L+1                                                    MLR 4310
         IF (JVOID(JV+1)) 230,260,260                                MLR 4320
     250 NOGOOD(L+1)=IABS(JVOID(JV))                                 MLR 4330
     260 CONTINUE                                                    MLR 4340
     270 NSKIP=IABS(NSKIP)                                           MLR 4350
         IF (NSKIP.NE.0) WRITE (6,960) VOTOED,(NOGOOD(J),J=1,LBAD)   MLR 4360
C             READ CONTROL CARD F                                    MLR 4370
         LOOK=0                                                      MLR 4380
         IF (NREAR) 290,300,280                                      MLR 4390
C             IF NREAR=1     READ SET OF SEARCH PARAMETERS.          MLR 4400
     280 NNXJ=NXV                                                    MLR 4410
         IF (IFWT.NE.0) NNXJ=NXV+1                                   MLR 4420
         READ (5,950) NWORDS,LOCY,(LOCX(J),J=1,NNXJ)                 MLR 4430
     290 WRITE (6,960) SEARCH,NWORDS,LOCY,(LOCX(J),J=1,NNXJ)         MLR 4440
         LOOK=1                                                      MLR 4450
     300 CONTINUE                                                    MLR 4460
C             READ CONTROL CARD G                                    MLR 4470
C                  READ TRANSFORMATIONS                              MLR 4480
         IF (IFTRA.GT.0) READ (5,930) (CONST(I),NBRTRA(I),I=1,NTOTAL)  MLR 4490
```

```
      IF (IFLIST.NE.0) GO TO 310                                        MLR 4500
      WRITE (6,990) ALPHA,D1,   (AX,J,J=1,NXV),(AY,J,J=1,NYV)
  310 NWT=NTOTAL                                                        MLR 4520
      IF (IFTRA.NE.0) WRITE (6,940) TRAN,(CONST(I),NBRTRA(I),I=1,NTOTAL)MLR 4530
C         READ CONTROL CARD H       NAME OF VARIABLES                   MLR 4540
      IF( IFNAME ) 315,314,312                                          MLR 4550
  312 DO 313 J=1,2                                                      MLR 4560
      DO 313 L=1,NTOTAL                                                 MLR 4570
  313 VNAME(J,L) = BLANK                                                MLR 4580
      GO TO 315                                                         MLR 4590
  314 READ(5,985) ( (VNAME(J,L), J=1,2), L=1,NTOTAL )                   MLR 4600
  315 IF(IFLIST.EQ.0)WRITE(6,986) ((VNAME(J,L),J=1,2),L=1,NTOTAL)       MLR 4610
C                                                                       MLR 4620
C         READ CONTROL CARD I   ( VARIABLE FORMAT )                     MLR 4630
      IF (NFMT.GT.0) READ (5,970) FMT                                   MLR 4640
      IF (NFMT.NE.0) WRITE (6,970) FORAMT,FMT                           MLR 4650
      IF (IFWT.NE.0) NWT=NTOTAL+1                                       MLR 4660
      NSKIP=IABS(NSKIP)                                                 MLR 4670
      NOAD=0                                                            MLR 4680
      YCONST=CONST(NEW)                                                 MLR 4690
      YTRA(1) = ACT                                                     MLR 4700
      YTRA(2) = CAL                                                     MLR 4790
      J = NBRTRA(NEW)                                                   MLR 4740
      NYTRA=0                                                           MLR 4700
      IF( IFTRA.EQ.0 .OR.  J.EQ.0 )  GO TO 316
      IF (NBRTRA(NEW).EQ.7) NYTRA=-1                                    MLR 4710
      IF (NBRTRA(NEW).EQ.8) NYTRA=+1                                    MLR 4720
      YTRA(1) = ATRA(1,J)                                               MLR 4750
      YTRA(2) = ATRA(2,J)                                               MLR 4760
  316 CONTINUE                                                          MLR 4010
      JDATA = 0
C                                                                       MLR 4020
C.......................................................................
C                                                                       MLR 4030
      DO 520 N=1,NDATA                                                  MLR 4040
      IF (NREAD) 370,370,320                                            MLR 4050
  320 IF (LOCK) 330,330,350                                             MLR 4060
  330 READ(NTAPE)   ( POINT(J),J=1,NWTT )
      IF( EOF(NTAPE) )  800, 420
  350 READ(NTAPE)   ( XDATA(J), J=1,NWORDS )
      IF( EOF(NTAPE) )  800, 400
C                                                                       MLR 5010
  370 IF (LOCK) 380,380,390                                             MLR 5020
  380 READ (5,FMT) (POINT(J),J=1,NWTT)                                  MLR 5030
      IF( EOF(5) )  800,385
  385 IF (NWRITE.NE.0) WRITE (10) (POINT(J),J=1,NWTT)                   MLR 5040
      GO TO 420                                                         MLR 5050
  390 READ (5,FMT) (XDATA(J),J=1,NWORDS)                                MLR 5060
      IF( EOF(5) )  800,395
  395 IF (NWRITE.NE.0) WRITE (10) (XDATA(J),J=1,NWORDS)                 MLR 5070
  400 DO 410 J=1,NNXJ                                                   MLR 5080
      JLOC=LOCX(J)                                                      MLR 5090
  410 POINT(J)=XDATA(JLOC)                                              MLR 5100
      POINT(NWT)=POINT(NNXJ)                                            MLR 5110
      POINT(NEW)=XDATA(LOCY)                                            MLR 5120
C                                                                       MLR 5130
C         CHECK FOR END OF DATA INDICATOR                               MLR 5140
```

```
  420 IF (POINT(1).EQ.99999999.) GO TO 530                            MLR 5150
      IF (IFSUB.GT.0) CALL EQUAT (IFSUB,POINT)                        MLR 5170
      IF (IFLIST.EQ.0) WRITE (6,910) N,(POINT(J),J=1,NWT)             MLR 5180
C         THROW AWAY BLANK DATA  ( BLANK = -0. )
      IFBLK = 0
      IF( IFBLK .NE. 0 ) GO TO 428
      DO 425 J=1,NXV
      IF( POINT(J) .NE. 0. ) GO TO 425
      IF( SIGN(+1., POINT(J) ) )  427,425,425
  425 CONTINUE
      IF( POINT(NEW) .NE. 0. ) GO TO 428
      IF( SIGN(+1., POINT(NEW) )) 427,428,428
  427 WRITE(6,1105)  N
      GO TO 520
C
  428 JDATA = JDATA + 1
      IF (IFTRA.EQ.0) GO TO 430                                       MLR 5190
      CALL CHANGE (POINT,NBRTRA,CONST,NTOTAL)                         MLR 5200
      IF (IFLIST.EQ.0) WRITE (6,920) N,(POINT(J),J=1,NWT)             MLR 5210
C                                                                     MLR 5220
  430 CONTINUE                                                        MLR 5230
                                                                      MLR 5240
      POINT(NOVAR)=POINT(NEW)                                         MLR 5250
      WHT=1.0                                                         MLR 5260
      IF (IFWT.NE.0) WHT=POINT(NWT)                                   MLR 5270
      POINT(NBRXYW)=WHT                                               MLR 5280
      IF (NCROSS.EQ.0) GO TO 450                                      MLR 5290
      IF( NCROSS .GT. 1 ) GO TO 445
C                  CROSS PRODUCTS ARE USED AS INDEPENDENT VARIABLES   MLR 5300
      L=NBRXY                                                         MLR 5310
      DO 440 I=2,NBRXY                                                MLR 5320
      DO 440 J=I,NBRXY                                                MLR 5330
      POINT(L)=POINT(I-1)*POINT(J-1)                                  MLR 5340
  440 L=L+1                                                           MLR 5350
      GO TO 450
C         GENERATE POWERS FOR POLYNOMIAL
  445 DO 446 J=2,NCROSS
  446 POINT(J) = POINT(J-1) * POINT(1)
  450 IF (IFCNST.LT.0) POINT(NOVAR-1)=1.0                             MLR 5360
      IF (IFBACK.EQ.0) GO TO 480                                      MLR 5370
      IF (NTAPE9.EQ.0) GO TO 470                                      MLR 5380
C         STORE IN STRING IF DATA POINTS * VARIABLES LESS THAN 3000   MLR 5390
      DO 460 J=1,NBRXYW                                               MLR 5400
      JJ=NBRXYW*(N-1)+J                                               MLR 5410
  460 STRING(JJ)=POINT(J)                                             MLR 5420
      GO TO 480                                                       MLR 5430
C     STORE DATA ON TAPE 9 IF DATA POINTS * VARIABLES EXCEED 3000     MLR 5440
  470 WRITE (9) (POINT(K),K=1,NOVAR),WHT                              MLR 5450
C                                                                     MLR 5460
  480 CONTINUE                                                        MLR 5470
      IF (NSKIP.EQ.0) GO TO 500                                       MLR 5480
C         CHECK TO SEE IF POINT IS TO BE DELETED FROM REGRESSION      MLR 5490
      DO 490 J=1,LBAD,2                                               MLR 5500
      IF (N.LT.NOGOOD(J).OR.N.GT.NOGOOD(J+1)) GO TO 490               MLR 5510
      NBAD=NBAD+1                                                     MLR 5520
      GO TO 520                                                       MLR 5530
  490 CONTINUE                                                        MLR 5540
```

```
      C                                                                          NLR 5550
        500 DO 510 I=1,NOVAR                                                     NLR 5560
      C         SUM X(I)                                                         NLR 5570
            A(I,NBRXYN)=A(I,NBRXYN)+POINT(I)*NHT                                 NLR 5580
            DO 510 J=I,NOVAR                                                     NLR 5590
      C         SUM X(I)*X(J)                                                    NLR 5600
        510 A(I,J)=A(I,J)+POINT(I)*POINT(J)*NHT                                  NLR 5610
            A(NBRXYN,NBRXYN)=A(NBRXYN,NBRXYN)+NHT                                NLR 5620
        520 CONTINUE                                                             NLR 5630
      C                                                                          NLR 5640
      C     ******************************************************************NLR 5650
      C                                                                          NLR 5660
        530 NDATA=JDATA-NBAD                                                     NLR 5670
            DEFRM=NDATA                                                          NLR 5680
            DENOM=A(NBRXYN,NBRXYN)-1.0                                           NLR 5690
            IF (IFWT.NE.0) DENOM=DENOM+1.0                                       NLR 5700
            IF (NTAPE9.EQ.0) RENINT 9                                            NLR 5710
            IF (NRENB5.EQ.0) RENINT 11                                           NLR 5720
      C                                                                          NLR 5730
            WRITE (6,1000) NPROB,ALPHA,NDATA,NOVAR,A(NBRXYN,NBRXYN),D1
            K=2                                                                  NLR 5750
            IF (IFTRA.EQ.0) K=1                                                  NLR 5760
            WRITE (6,860) (BLANK,J=1,K)                                          NLR 5770
            WRITE (6,870)                                                        NLR 5780
            IX1=60                                                               NLR 5790
            IX2=0                                                                NLR 5800
            IXP = 1
      C                                                                          NLR 5810
            DO 590 J=1,NOVAR                                                     NLR 5820
            YMEAN=A(J,NBRXYN)/A(NBRXYN,NBRXYN)                                   NLR 5830
            STDEV=SQRT((A(J,J)-A(J,NBRXYN)*YMEAN)/DENOM)                         NLR 5840
            TRA(1)=BLANK                                                         NLR 5850
            TRA(2)=BLANK                                                         NLR 5860
            V(1,J)= VNAME(1,J)                                                   NLR 5870
            V(2,J) = VNAME(2,J)                                                  NLR 5880
            L=J                                                                  NLR 5890
            K=2                                                                  NLR 5900
            IF (J.GT.NXV) GO TO 560                                              NLR 5910
        540 IF (IFTRA.EQ.0) GO TO 580                                            NLR 5920
            I=NBRTPA(L)                                                          NLR 5930
            IF (I.LE.0) GO TO 580                                                NLR 5940
            K=3                                                                  NLR 5950
            TRA(3)=CONST(L)                                                      NLR 5960
        550 TRA(1)=ATRA(1,I)                                                     NLR 5970
            TRA(2)=ATRA(2,I)                                                     NLR 5980
            GO TO 580                                                            NLR 5990
        560 I=17                                                                 NLR 6000
            V(1,J) = 6HCONST.
            V(2,J) = 5H TERM
            IF (IFCNST.LT.0.AND.J.EQ.NOVAR-1) GO TO 550                          NLR 6030
            L=NXV+INDEXY                                                         NLR 6040
            V(1,J) = VNAME(1,L)                                                  NLR 6050
            V(2,J) = VNAME(2,L)                                                  NLR 6060
            IF (J.EQ.NOVAR) GO TO 540                                            NLR 6070
            IF( NCROSS .GE. 2 )  GO TO 575
      C         CROSS PRODUCTS                                                   NLR 6080
            IX1=IX1+1                                                            NLR 6090
```

```
      IF (IX1.LE.NXV) GO TO 570                                         MLR 6100
      IX2=IX2+1                                                         MLR 6110
      IX1=IX2                                                           MLR 6120
  570 WRITE (6,880) J,YMEAN,STDEV,IX2,IX1                               MLR 6130
      V(1,J) = CROSS(IX2)                                               MLR 6140
      V(2,J) = CROSS(IX1)                                               MLR 6150
      GO TO 590                                                         MLR 6160
C         POLYNOMIALS
  575 V(1,J) = V(1,1)
      IXP = IXP + 1
      ENCODE(6,1140,V(2,J))  IXP
  580 IF(J.EQ.NOVAR) GO TO 590                                          MLR 6170
      WRITE(6,890) V(1,J),V(2,J),J, YMEAN, STDEV, (TRA(L),L=1,K)        MLR 6180
  590 CONTINUE                                                          MLR 6190
      J = NOVAR
      WRITE(6,900) V(1,J),V(2,J),   YMEAN, STDEV, (TRA(L),L=1,K)        MLR 6200
C                                                                       MLR 6210
      IF (IFSUMS.EQ.0) CALL PRTSUM                                      MLR 6220
      IF (IFCNST.NE.0) GO TO 600                                        MLR 6230
      CALL RESID                                                        MLR 6240
      DEFRM=DEFRM-1.0                                                   MLR 6250
  600 CALL CORREL                                                       MLR 6260
      CALL SLITET(1,LIGHT)                                              MLR 6270
      IF( LIGHT.EQ.1 )  GO TO 10                                        MLR 6280
      NDATA=JDATA                                                       MLR 6290
      NSKIP=NBAD                                                        MLR 6300
      IF(MAXSTP.EQ.999) WRITE(6,1060)                                   MLR 6310
C     ******************************************************************MLR 6320
C                                                                       MLR 6330
  605 LPATH = 1                                                         MLR 6340
      KPATH = 1                                                         MLR 6350
      IF (NBRNOW.GT.0) GO TO 710                                        MLR 6360
  610 IF (A(NOVAR,NOVAR).GT.0.0) GO TO 620                              MLR 6370
      WRITE (6,1030) A(NOVAR,NOVAR)                                     MLR 6380
      GO TO 750                                                         MLR 6400
C                                                                       MLR 6410
  615 JP = 0                                                            MLR 6420
      NBRPVR = NBRNOW                                                   MLR 6430
      NBRNOW = NBRNOW - 1                                               MLR 6440
      DO 617 J=1,NBRPVR                                                 MLR 6450
      CALL PACK( NBRPVR, J, 0, 1 )                                      MLR 6460
      IF( INDEX(J) .EQ. KVAR )  GO TO 617
      JP = JP + 1                                                       MLR 6480
      CALL PACK( NBRNOW, JP, INDEX(J), 1 )                              MLR 6490
  617 CONTINUE                                                          MLR 6500
      DO 618 I=1,NOVAR                                                  MLR 6510
      DO 618 J=1,NOVAR                                                  MLR 6520
  618 A(I,J) = SIMCOR(I,J)                                              MLR 6530
      GO TO 605                                                         MLR 6540
C                                                                       MLR 6550
  620 CALL ACOTO                                                        MLR 6560
      CALL SLITET (1,LIGHT)                                             MLR 6570
      GO TO (750,630), LIGHT                                            MLR 6580
  630 GO TO (640,720,690), LPATH                                        MLR 6590
C                                                                       MLR 6600
  640 TEST(NBRNOW)=A(NOVAR,NOVAR)                                       MLR 6610
  650 CALL OUTPUT                                                       MLR 6620
```

```
PROGRAM MLR        74/74    OPT=1                               FTN 4.2+74270        11/22/74  10.10.21.

            CALL SLITET (2,LIGHT)                                                     MLR 6630
            IF( LIGHT .EQ. 1 ) GO TO 615                                              MLR 6640
            NSTEP=NSTEP+1                                                             MLR 6650
            IF (NSTEP.GT.MAXSTP) GO TO 750                                            MLR 6660
            IF (NBRNOW.GE.MAXREG) GO TO 750                                           MLR 6670
            CALL SLITET (1,LIGHT)                                                     MLR 6680
            GO TO (750,660), LIGHT                                                    MLR 6690
        660 GO TO (680,670,10), KPATH                                                 MLR 6700
        670 KPATH=1                                                                   MLR 6710
      C          SEE IF VARIABLE CAN BE REMOVED
        680 IF( NBRNOW.LE.2  .AND.  JF.NE.0 )  GO TO 700
            CALL REMOVE                                                               MLR 6730
            IF( MODEL.EQ.0 )  GO TO 700
      C          VARIABLE WAS REMOVED
            LPATH=1                                                                   MLR 6750
        690 CALL MATRIX                                                               MLR 6760
            GO TO (650,740,700), LPATH                                                MLR 6770
        700 IF (NBRNOW.LT.NBRX) GO TO 610                                             MLR 6780
            WRITE (6,1020)                                                            MLR 6790
            GO TO 750                                                                 MLR 6800
        710 KPATH=2                                                                   MLR 6810
        720 L=1                                                                       MLR 6820
            LPATH=2                                                                   MLR 6830
        730 CONTINUE                                                                  MLR 6840
            CALL PACK (NORNOW,L,KVAR,2)
            GO TO 690                                                                 MLR 6860
      C      ******************** TRY ADJUNCTION *************************** MLR 6870
        740 L=L+1                                                                     MLR 6880
            IF (L.LE.NBRNOW) GO TO 730                                                MLR 6890
            GO TO (700,640,780), KPATH                                                MLR 6900
      C                                                                               MLR 6910
      C          SUMMARY                                                              MLR 6920
        750 WRITE(6,1040) ALPHA,NPROB                                                 MLR 6930
            IF (IFBACK.EQ.999) IFBACK=NSTEP                                           MLR 6940
            IF (IFBACK.EQ.999) TFOACK=1                                               MLR 6950
            IF (MAXSTP.EQ.999) MAXSTP=998                                             MLR 6960
            DO 757 J=1,NBRNOW                                                         MLR 6970
            DO 755 L=1,J                                                              MLR 6980
        755 CALL PACK(J,L,INDEX(L),2)                                                 MLR 6990
            IF( INDEX(1).LE.0 ) GO TO 757                                             MLR 7000
            WRITE(6,1070)J,KSTEP(J),STOERR(J),CORSCR(J), (INDEX(L),L=1,J)             MLR 7010
        757 CONTINUE                                                                  MLR 7020
            KPATH=3                                                                   MLR 7030
            NBRNOW=MINVAR                                                             MLR 7040
            NSUMRY=1                                                                  MLR 7050
        760 IF( NBRNOW.GT.MAXVAR ) GO TO 795
            CALL PACK (NBRNOW,1,J,2)                                                  MLR 7070
            IF (J.LE.0) GO TO 790                                                     MLR 7080
            DO 770 I=1,NOVAR                                                          MLR 7090
            DO 770 J=1,NOVAR                                                          MLR 7100
        770 A(I,J)=SIMCOR(I,J)                                                        MLR 7110
            GO TO 720                                                                 MLR 7120
        780 CALL OUTPUT                                                               MLR 7130
        790 NBRNOW=NBRNOW+1                                                           MLR 7140
            GO TO 760                                                                 MLR 7150
      C          COMPUTE CP SECONDS FOR PROBLEM
        795 PEND   = SECOND( PEND )
```

```
          PSTART = PEND - PSTART
          WRITE(6,1100) PSTART
          PSTART = PEND
          GO TO 10
      800 WRITE (6,1050) JDATA,(POINT(J),J=1,NWTT)                  MLR 7160
          GO TO 531
      805 WRITE(6,1090)
CCC       STOP
C                                                                   MLR 7190
C                                                                   MLR 7200
C                                                                   MLR 7210
      810 FORMAT (47H1 NONSIMPLE STEPWISE MULTIPLE LINEAR REGRESSION45X23HAEMLR 7220
         1MY COMPUTATION CENTER/92X20HARMY MISSILE COMMAND/92X25HREDSTONE ARMLR 7230
         2SENAL, ALABAMA)                                           MLR 7240
      820 FORMAT (37H0 TOTAL NUMBER OF VARIABLES TOO LARGEI6)       MLR 7250
      830 FORMAT (16A5)                                             MLR 7260
      840 FORMAT (/3X,16A5)                                         MLR 7270
      850 FORMAT (47H  CROSS PRODUCTS DELETED, AS THERE ARE TOO MANY//)    MLR 7280
      860 FORMAT (20X35HVAR    WTED AVERAGE     STAND. DEV. ,2A1,3X,25HTRANSMLR 7290
         1FORMATION   CONSTANT)                                     MLR 7300
      870 FORMAT (1X)                                               MLR 7310
      890 FORMAT (7X,2A6,I3    ,1X,2E16.6,6X,A10,A2,E16.4)
      880 FORMAT (19X    ,I3    ,1X,2E16.6,6X,2HX(I2,6H) *.X(I2,1H) )    MLR 7310
      900 FORMAT (7X,2A6,3H  Y,1X,2E16.6,6X,A10,A2,E16.4)
      910 FORMAT (I9,1X,10F12.3/(10X,10F12.3))                      MLR 7350
      920 FORMAT (5X,1H(I3,1H),10F12.3/(10X,10F12.3))               MLR 7360
      930 FORMAT (7(F8.0,I2))                                       MLR 7370
      940 FORMAT (1H ,3XA6,10(F10.3,I2)/(10X,10(F10.3,I2)))         MLR 7380
      950 FORMAT (14I5)                                             MLR 7390
      960 FORMAT (1HJ,A6,24I5/(7X24I5))                             MLR 7400
      970 FORMAT( 11A10 )
      985 FORMAT( 7(A6,A4) )                                        MLR 7430
      986 FORMAT( 12X20A6 )                                         MLR 7440
      990 FORMAT (34H1  * *  I N P U T     D A T A  * *5X16A5,2XA10  //E(9X,
         110(7XA1,1H(I2,1H))/))                                     MLR 7460
     1000 FORMAT (37H1STEPWISE REGRESSION PROBLEM NUMBER  I5,10X16A5/23H NUMMLR 7470
         1BER OF OBSERVATIONS14X,I5/20H NUMBER OF VARIABLES17XI5/30H WEIGHTEMLR 7480
         2D DEGREES OF FREEDOM  F12.3,78XA10//)
     1010 FORMAT (//35H  P R O G R A M     C O N T R O L S///10H   NFROB =IMLR 7500
         15,10H   NXV   =I5,10H   NYV   =I5,10H   INDEXY=I5,10H   NDATA =I5,MLR 7510
         210H   IOEM  =I5,10H   INTYPE=I5//10H   NREAR =I5,10H   MAXSTF=I5,1MLR 7520
         30H   IFBACK=I5,10H   MSTART=I5,10H   MINSUM=I5,10H   MAXSUM=I5,10HMLR 7530
         4   MAXREG=I5//10H   IFWT  =I5,10H   IFCAST=I5,10H   IFLIST=I5,10H MLR 7540
         5   IFSUPS=I5,10H   IFRES =I5,10H   IFCORR=I5,10H   NCROS =I5//10H  MLR 7550
         6 IFTRA =I5,10H   NVOID =I5,10H   IFSUB =I5,10H   NFMT   I5,10H   IMLR 7560
         7FPNCH=I5,10H   IFDATE=I5,10H   IFNAME=,I5)                MLR 7570
     1020 FORMAT (51H0 ***************** VARIABLES EXHAUSTED ************/)  MLR 7580
     1030 FORMAT (80H0************* PROBLEM TERMINATED, SUM OF SQUARES IS NOMLR 7590
         1N-POSITIVE **************E13.5/)                          MLR 7600
     1050 FORMAT (//34H END OF FILE REACHED AFTER READINGI5,8H POINTS./20HOLMLR 7610
         1AST OBSERVATION IS//(1X,8E16.8))                          MLR 7620
     1060 FORMAT(33H1MULTIPLE LINEAR REGRESSION STEPS)              MLR 7630
     1040 FORMAT(34H1SUMMARY OF BEST SETS OF VARIABLES,4X16A5,9H  PROBLEMI6/MLR 7640
         */25H  M STEP    STD ERR   RSQ / )                         MLR 7650
     1070 FORMAT(1X,I2,I4,F11.3,F9.6,1X,35I3/28X,35I3)              MLR 7660
     1080 FORMAT( 6A10//8A10/8A10//40( A4,2X,3F10.3/ ) )
     1090 FORMAT( 1H1,50X,10HEND OF JOB )
```

```
PROGRAM MLR        74/74    OPT=1                    FTN 4.2+74278      11/22/74  18.18.21.

       1100 FORMAT(1H1,F10.3/20(/1H-,60(2H* )))
       1105 FORMAT(6H POINT I4,26H DELETED DUE TO BLANK DATA )
       1110 FORMAT(25H0CROSS PRODUCTS GENERATED )
       1120 FORMAT(41H0POWERS GENERATED FOR POLYNOMIAL OF ORDER I3)
       1130 FORMAT(53H0ONLY ONE INDEPENDENT VARIABLE ALLOWED FOR POLYNOMIAL)
       1140 FORMAT(4H ** ,I2)
      C
            END                                                       MLR 7730
```

```
      SUBROUTINE OUTPUT                                                 AUTP   0
      COMMON SIGMA(60),A(52,52),SIMCOR(52,52),AVG(60),TEST(60)          AUTP  10
      COMMON PCINT(60),STRING(3000),INDPAC(30,30),INDEXP(61)            AUTP  20
      COMMON INDEX(60),NOUT(60),KSTEP(60),ALPHA(16),YMEAN,IDEN,IFAVE    AUTP  30
      COMMON MAXSTP,IFPNCH,NSUMRY,NSKIP,NTAPE9,NEW                      AUTP  40
      COMMON NCVAR,NBRNOW,NOSTEP,NDATA,NBRXY,NBRX,LPATH,DEFRM,KVAR      AUTP  50
      COMMON IFBACK,IFCNST,IFCORR,NPROB,NBRPVR,TOL,NCDEL,JF             AUTP  60
      COMMON INDEXY,LBAD,NOGOOD(28)                                     AUTP  70
      COMMON IFWT,YCONST,NYTRA,V(2,51),YTRA(2)                          AUTP  80
      COMMON  STOERR(50),CORSQR(50)                                     AUTP  90
      REAL            A,SIMCOR,SIGMA,AVG,TEST                           AUTP 100
C                                                                       AUTP 110
      DIMENSION  COEFF(61), ABC(5)                                      AUTP 120
      REAL            COEFF,CONST                                       AUTP 130
      REAL            SUMSQ,TSS,SIGY2,SIGY                              AUTP 140
      REAL            YPRED,YOBS,DEV,RSO,SQREG,SQREG2                   AUTP 150
      REAL             DEVSQ,CHISO,SUMSCU,CHISQL,DEVU,YO,YC
      DATA BLANK/1H /,VOID/6HVOIDED/,CHECK/6HREVIEW/                    AUTP 170
      DATA  ACTUAL/6HACTUAL/                                            AUTP 180
C         NBRNOW = NUMBER OF COEFFICIENTS FOR PRESENT  ECUATION         AUTP 190
C         INDEX  = INDEX OF PRESENT  EQUATION                           AUTP 200
C         NSUMRY = 0 FOR BUILDING PHASE. = 1 FOR SUMMARY PHASE.         AUTP 210
      IF( NBRNOW.EQ.1 )   JF = 1
      KPATH=1                                                           AUTP 220
      IF (NSKIP.NE.0) KPATH=2                                           AUTP 230
      WMT=1.0                                                           AUTP 240
      TSS = SIGMA(NDVAR)*SIGMA(NDVAR)                                   AUTP 250
      CALL SLITET (1,LIGHT)                                             AUTP 260
      GO TO (10,20), LIGHT                                              AUTP 270
   10 CALL SLITE (1)                                                    AUTP 280
      GO TO 30                                                          AUTP 290
   20 NOSTEP=NOSTEP+1                                                   AUTP 300
      IF (NSUMRY.EQ.1) NOSTEP=KSTEP(NBRNOW)                             AUTP 310
   30 DO 40 J=1,60                                                      AUTP 320
   40 NOUT(J)=0                                                         AUTP 330
      DO 50 J=1,NBRNOW                                                  AUTP 340
      CALL PACK (NBRNOW,J,I,2)                                          AUTP 350
      INDEX(J)=I                                                        AUTP 360
      NOUT(I)=1                                                         AUTP 370
C           BETA = A(I,NOVAR)                                           AUTP 380
   50 COEFF(J)=A(I,NOVAR)*SIGMA(NOVAR)/SIGMA(I)                         AUTP 390
      IF (IFCNST.EQ.0) GO TO 60                                         AUTP 400
      CONST=0.0                                                         AUTP 410
      GO TO 80                                                          AUTP 420
   60 CONST=AVG(NOVAR)                                                  AUTP 430
      DO 70 I=1,NBRNOW                                                  AUTP 440
      J=INDEX(I)                                                        AUTP 450
   70 CONST=CONST-(COEFF(I)*AVG(J))                                     AUTP 460
   80 SUMSQ = A(NOVAR,NOVAR) * TSS                                      AUTP 470
      XVAR=NBRNOW                                                       AUTP 480
      DEFR=DEFRM-XVAR                                                   AUTP 490
      NDEFR=DEFR                                                        AUTP 500
      NDEFRM=DEFRM                                                      AUTP 510
C.......................................................................

      IF (A(NOVAR,NOVAR).LT.0.0) A(NOVAR,NOVAR)=0.0D0                   AUTP1460
CC    SUMSQ=DABS(SUMSQ)                                                 AUTP1470
```

```
            SUMSQ= ABS(SUMSQ)
            IF (NDEFR.GT.0) GO TO 240                                       AUTP1480
            WRITE (6,370) NOSTEP                                            AUTP1490
            SIGY=0.0                                                        AUTP1500
            SIGPCT=0.0                                                      AUTP1510
            REGRCO=0.0                                                      AUTP1520
            CALL SLITE (1)                                                  AUTP1530
            STDERR(NBRNOW) = 0.0                                            AUTP1540
            CORSQR(NBRNOW) = 0.0                                            AUTP1550
            IF (MAXSTP.EQ.999)  GO TO 999
            GO TO 260                                                       AUTP1570
        240 CONTINUE                                                        AUTP1580
            SIGY2=SUMSQ/DEFR                                                AUTP1590
      CC    SIGY=DSQRT(SIGY2)                                               AUTP1600
            SIGY = SQRT ( SIGY2 )
            RSQ=1.0D0-A(NOVAR,NOVAR)                                        AUTP1610
            IF( NOSTEP .NE. KSTEP(NBRNOW) )  GO TO 245                      AUTP1620
            STDERR(NBRNOW) = SIGY                                           AUTP1630
            CORSQR(NBRNOW) = RSQ                                            AUTP1640
        245 CONTINUE                                                        AUTP1650
            IF (MAXSTP.EQ.999)  GO TO 999
            AYY=A(NOVAR,NOVAR)                                              AUTP1670
            TR=1.0                                                          AUTP1680
            REGRCO=SQRT(RSQ)                                                AUTP1690
            R2=RSQ/XVAR                                                     AUTP1700
            V2=A(NOVAR,NOVAR)/DEFR                                          AUTP1710
            FTEST=R2/V2                                                     AUTP1720
            SQREG=RSQ*TSS                                                   AUTP1730
            SQREG2=SQREG/XVAR                                               AUTP1740
            DATA CORREC,TED/6HCORREC,3NTED/,ORI,GIN/6H ORIGI,1NN/           AUTP1750
            BASE1=CORREC                                                    AUTP1760
            BASE2=TED                                                       AUTP1770
            IF (IFCNST.EQ.0) GO TO 250                                      AUTP1780
            BASE1=ORI                                                       AUTP1790
            BASE2=GIN                                                       AUTP1800
        250 WRITE (6,440) NOSTEP,ALPHA,NPRCB,BASE1,BASE2,NDEFRM,TSS,TR,NERNCW,AUTP1810
           1SQREG,SQREG2,RSQ,R2,FTEST,NDEFR,SUMSQ,SIGY2,AYY,V2              AUTP1820
            SIGPCT=ABS(SIGY/YMEAN*100.0)                                    AUTP1830
      C                                                                     AUTP1840
        260 WRITE(6,380) SIGY,V(1,NOVAR),V(2,NOVAR),SIGPCT,REGRCO           AUTP1850
            DO 290 J=1,NBRX                                                 AUTP1860
      CC    IF (A(J,J)-TOL) 270,270,280                                     AUTP1870
      CC270 POINT(J)=0.99E22                                                AUTP1880
      CC    GO TO 290                                                       AUTP1890
        280 POINT(J)=A(J,NOVAR)*A(NOVAR,J)/A(J,J)                           AUTP1900
        290 CONTINUE                                                        AUTP1910
      C                                                                     AUTP1920
            WRITE (6,400)                                                   AUTP1930
            KK=0                                                            AUTP1940
            IF (IFPNCH.NE.0.AND.NSUMRY.EQ.1) WRITE (7,480)  KK,CONST,NFRCE,NBRNAUTP1950
           1OW,INDEXY,NOSTEP,IFPNCH,SIGPCT,REGRCO                           AUTP1960
            IF (IFCNST.EQ.0) WRITE (6,390) CONST                            AUTP1970
      C                                                                     AUTP1980
      C     **** LIST COEFFICIENTS                                          AUTP1990
            DO 310 J=1,NBRNOW                                               AUTP2000
            I=INDEX(J)                                                      AUTP2010
            IF (IFPNCH.EQ.0.OR.NSUMRY.EQ.0) GO TO 300                       AUTP2020
```

```
SUBROUTINE OUTPUT     74/74    OPT=1                       FTN 4.2+74278      11/22/76  16.16.??.

          WRITE (7,480) I,COEFF(J),NPROB,NBRNOW,INOEXY,NCSTEP,IFPNCH,SIGFCT,AUTP2030
         1REGRCO                                                                 AUTP2040
      300 SEB=SQRT(ABS(A(I,I)))*SIGY/SIGMA(I)                                    AUTP2050
          CT=COEFF(J)/SEB                                                        AUTP2060
          IF( SEB .EQ. 0.0 )  CT = 0.0                                           AUTP2070
          F = CT * CT
          WRITE(6,410) V(1,I),V(2,I),I,COEFF(J),SEB,CT,F,A(I,NOVAR),POINT(I)
          IF( POINT(I).LE.0.0 )  GO TO 310                                       AUTP2090
          WRITE(6,510)                                                           AUTP2100
          CALL SLITE (2)                                                         AUTP2110
          KVAR = I                                                               AUTP2120
      310 CONTINUE                                                               AUTP2130
    C                                                                            AUTP2140
          IF (NBRNOW.EQ.NBRX) GO TO 360                                          AUTP2150
          IF (NOEFR.LE.0) GO TO 360                                              AUTP2160
          WRITE (6,420)                                                          AUTP2170
          NP=0                                                                   AUTP2180
          CT = NOEFR - 1
          DO 350 I=1,NBRX                                                        AUTP2190
          IF( NOUT(I) )  350, 315, 350
      315 F = ( CT * POINT(I) ) / ( AYY - POINT(I) )
          PAR=POINT(I)/AYY                                                       AUTP2200
          IF( A(I,I).LE.TOL )  POINT(I) =3.3333333E33                            AUTP2210
      320 IF (NP) 340,330,340                                                    AUTP2230
      330 IH=I                                                                   AUTP2240
          FH = F
          RPAR=PAR                                                               AUTP2250
          SSN=A(I,I)                                                             AUTP2260
          DELT=POINT(I)                                                          AUTP2270
          NP=1                                                                   AUTP2280
          GO TO 350                                                              AUTP2290
      340 WRITE(6,430) IH,RPAR,SSN,DELT,FH,  I,PAR,A(I,I)-POINT(I),F
          NP=0                                                                   AUTP2310
      350 CONTINUE                                                               AUTP2320
          IF( NP.NE.0 )  WRITE(6,430) IH,RPAR,SSN,DELT,FH
      360 CONTINUE

    C..........................................................................

    C     **** COMPUTE BACK SOLUTION                                             AUTP 550
          IF (MAXSTP.EQ.999) GO TO 999
          IF (IFBACK.EQ.0.OR.IFBACK.GT.NOSTEP) GO TO 999
    C                                                                            AUTP 560
          NABC = 1                                                               AUTP 570
          IF( IFWT.NE.0 ) NABC = 2                                               AUTP 580
          IF(NYTRA.NE.0) NABC = 5                                                AUTP 590
          SUMSQ=0.000                                                            AUTP 610
          CHISQ = 0.000                                                          AUTP 620
          SUMSQU = 0.000                                                         AUTP 630
          CHISQU = 0.000                                                         AUTP 640
          NONO = 0                                                               AUTP 650
          NDROP=0                                                                AUTP 660
          LINE=50                                                                AUTP 670
    C                                                                            AUTP 680
          DO 220 N=1,NDATA                                                       AUTP 690
          IF (NTAPE9.NE.0) GO TO 90                                              AUTP 700
          READ (9) (POINT(L),L=1,NOVAR),WHT                                      AUTP 710
```

```
         GO TO 110                                                      AUTP 720
      90 JJ=NBRXYH*(N-1)                                                AUTP 730
         DO 100 L=1,NOVAR                                               AUTP 740
         KK=JJ+L                                                        AUTP 750
     100 POINT(L)=STRING(KK)                                            AUTP 760
         JJ=NBRXYH*N                                                    AUTP 770
         WHT=STRING(JJ)                                                 AUTP 780
C                                                                       AUTP 790
     110 YPRED=CONST                                                    AUTP 800
         DO 120 I=1,NBRNOW                                              AUTP 810
         J=INDEX(I)                                                     AUTP 820
     120 YPRED=YPRED+COEFF(I)*POINT(J)                                  AUTP 830
C                                                                       AUTP 840
         YOBS=POINT(NOVAR)                                              AUTP 850
         DEV=YPRED-YOBS                                                 AUTP 860
         DEVN=DEV/SIGY                                                  AUTP 870
         IF( NDEFR .LE. 0 )  DEVN = 0.0                                 AUTP 880
         PCT = (DEV*100.0)/YOBS
         GOOD=BLANK                                                     AUTP 890
         IF (ABS(DEVN).GT.3.5) GOOD=CHECK                               AUTP 910
         GO TO (150,130), KPATH                                         AUTP 920
     130 DO 140 J=1,LBAD,2                                              AUTP 930
         IF (N.LT.NOGOOD(J).OR.N.GT.NOGOOD(J+1)) GO TO 140              AUTP 940
         GOOD=VOID                                                      AUTP 950
         NBAD=NBAD+1                                                    AUTP 960
         IF (NBAD.EQ.NSKIP) KPATH=1                                     AUTP 970
         GO TO 160                                                      AUTP 980
     140 CONTINUE                                                       AUTP 990
     150 DEVSQ = (DEV*DEV)*WHT                                          AUTP1000
         SUMSQ = SUMSQ + DEVSQ                                          AUTP1010
         CHISQ = CHISQ + DEVSQ/YPRED                                    AUTP1020
     160 LINE=LINE+1                                                    AUTP1030
         M = NAEC                                                       AUTP1040
         ABC(1) = GOOD                                                  AUTP1050
         ABC(2) = WHT                                                   AUTP1060
         IF (NYTRA) 170,190,180                                         AUTP1070
     170 IF (YOBS.GT.15..OR.YPRED.GT.15.) GO TO 185                     AUTP1080
         YO = EXP ( YOBS ) - YCONST
         YC = EXP ( YPRED ) - YCONST
         GO TO 200                                                      AUTP1110
     180 IF (YOBS.GT.8..OR.YPRED.GT.8.) GO TO 185                       AUTP1120
         YO = 10.0D0**YOBS  - YCONST                                    AUTP1130
         YC = 10.0D0**YPRED - YCONST                                    AUTP1140
     200 CONTINUE                                                       AUTP1150
         DEVU = YC - YO                                                 AUTP1160
         ABC(3) = YO                                                    AUTP1170
         ABC(4) = YC                                                    AUTP1180
         ABC(5) = DEVU                                                  AUTP1190
         IF( GOOD.EQ.VOID ) GO TO 190                                   AUTP1200
         DEVSQ = (DEVU*DEVU) * WHT                                      AUTP1210
         SUMSQU = SUMSQU + DEVSQ                                        AUTP1220
         CHISQU = CHISQU + DEVSQ/YC                                     AUTP1230
         GO TO 190                                                      AUTP1240
     185 NONO = 1                                                       AUTP1250
         M = 2                                                          AUTP1260
     190 CONTINUE                                                       AUTP1270
         IF (LINE.LE.50) GO TO 210                                      AUTP1280
```

SUBROUTINE OUTPUT   74/74   OPT=1   FTN 4.2+74270   11/22/74 18.18.37.

```
      WRITE(6,460)NOSTEP,NPROB,(V(J,NOVAR),J=1,2),(YTRA(J),J=1,2),      ALTP1290
     *     (BLANK, J=1,NABC)                                            AUTP1200
      LINE=1                                                            ALTP1210
  210 WRITE(6,470) GOOD,N,YOBS,YPREC,DEV,DEVN,PCT,(ABC(J),J=1,M)        ALTP1320
  220 CONTINUE                                                          ALTP1330

      SIGY = SQRT ( SUMSQ/DEFR )
      SIGYU = SQRT ( SUMSQU/DEFR )
      L=1                                                               AUTP1370
      IF(YTRA(1) .NE.ACTUAL) L=2                                        ALTP1380
      WRITE(6,490) SIGY, CHISQ, SUMSQ, (BLANK, J=1,L)                   ALTP1350
      IF( NABC.EQ.5 .AND. NCNO.EQ.0 ) WRITE(6,500) SIGYU,CHISQL,SUMSQL  AUTP1400
      IF (NTAPE9.EQ.0) REWIND 9                                         ALTP1410
      SUMSQ = A(NOVAR,NOVAR) * TSS                                      ALTP1430
C.......................................................................

  999 RETURN
C                                                                       ALTP2350
  370 FORMAT (29H1NO MORE DEGREES FREEDOM STEPI5/1H0,120(1H*))          ALTP2360
  380 FORMAT (20H0STANDARD ERROR OF Y F16.6,5X2A6/24H S.E. AS PERCENT OFAUTP2370
     * MEAN F12.6/24H CORRELATION COEFFICIENT F12.6 // )                ALTP2380
  390 FORMAT (7X,15HCONST. TERM   0,E16.8)                              AUTP2390
  400 FORMAT(20X,48HVAR    COEFFICIENT          STAND. ERROR   COEF/SE,8X,ALTP2400
     *1HF, 9X, 4HBETA, 7X,10HRSQ CHANGE )
  410 FORMAT(7X2A6,I3, E16.8,E19.6,F11.4,F11.2, 2F14.8 )
  420 FORMAT(1H-27X58HREGRESSION OF THE VARIABLE K ON THE SET OF VARIABLALTP2430
     1ES ABOVE//    2(5X,44HK     PARTIAL RSQ    NORMED SUM/SQ     DELTA RSQ,
     2 6X,4HF    ,5X ) )
  430 FORMAT(2(4XI2,F12.7,E17.7,E15.7,1XF8.3, 5X) )
  440 FORMAT (5H1STEPI4,9X,16A5,9H  PROBLEMI6/6H0ANOVA.21X,30H....... ORAUTP2470
     1IGINAL UNITS .......9X,26H.... CORRELATION FORM ..../8H  SOURCE13XALTP2480
     2,36HC.F.  SUM OF SQUARES    MEAN SQUARES9X,26HSUM SQUARES   MEAN SAUTP2490
     3QUARES11X,9HOVERALL F/7H0TOTAL(A6,A3,1H)I7,E17.8,20X,F15.8/11H REGAUTP2500
     4RESSIONI13.2E17.8,3X,2F15.8,F20.4/11H RESIDUAL  I13,2E17.8,3X,2F15AUTP2510
     5.8/)                                                              AUTP2520
  460 FORMAT (5H1STEPI4,10X,43HRESIDUAL ANALYSIS  ( ACTUAL VS CALCULATEDALTP2530
     1 ).9X,7HPROBLEMI6/1H0,22X,A6,A4/7X,16HOBSERVATION     A1C,A2, 11H
     2CALCULATED,8X,30HRESIDUAL     NOR DEV     PCT DEV.A3,A6,6HWEIGHT,3A1AUTP2550
     3,30H   ACTUAL       CALC.         DEV //)                         ALTP2560
  470 FORMAT (1X,A6,I6,F17.5,2F16.5,F10.3,F11.3,2XA6,F8.4,F12.2,2F11.2) ALTP2570
  480 FORMAT (I5,E20.8,5I5,F20.4,F10.7)                                 AUTP2580
  490 FORMAT (17H0STANDARD ERROR =F10.3,15H , CHI SQUARE =F11.4,2X,     ALTP2590
     *  7H   SSE =E15.8,2A1,34HFOR TRANSFORMED DEPENDENT VARIABLE )     AUTP2600
  500 FORMAT (17H0STANDARD ERROR = F10.3,15H , CHI SQUARE =F11.4,2X,    ALTP2610
     *  7H   SSE =E15.8,2X ,34HFOR RECONVERTED DEPENDENT VARIABLE )     ALTP2620
  510 FORMAT(1X,50(1H*),53H ILL-CONDITIONED SET --- RESULTS IN DOUBT. DOAUTP2630
     *NOT USE , 19(1H*) )                                               ALTP2640
      END                                                               ALTP2650
```

```
      SUBROUTINE PRTSUM                                                 SOME  10
C     S/R TO PRINT SUMS OF CROSS PRODUCTS                               SOME   8
C       RAW SUMS OF SQUARES AND CROSS PRODUCTS                          SOME  20
      COMMON SIGMA(60),A(52,52),SIMCOR(52,52),AVG(60),TEST(60)          SOME  30
      COMMON PCINT(60),STRING(3000),INOPAC(30,38),INDEXP(61)            SOME  40
      COMMON INDEX(60),NOUT(60),KSTEP(60),ALPHA(16),YMEAN,IOEN,IFAVE    SOME  50
      COMMON MAXSTP,IFPNCH,NSUMRY,NSKIP,NTAPE9,NEW                      SOME  60
      COMMON NOVAR,NBRNOW,NOSTEP,NOATA,NBRXYN,NBRX,LPATH,OEFRM,NVAF     SOME  70
      COMMON IFBACK,IFCNST,IFCORR,NPROB,NBRPVR,TOL,NOOEL,JF             SOME  80
      COMMON INDEXY,LBAO,NOGOOO(28)                                     SOME  90
      COMMON IFWT,YCONST,NYTRA,V(2,51),YTRA(2)                          SOME 100
      COMMON  STOERR(50),CORSQR(50)                                     SOME 110
      REAL            A,SIMCOR,SIGMA,AVG,TEST                           SOME 120
C                                                                       SOME 130
      OATA JB/1M /                                                      SOME 140
      WRITE (6,150)                                                     SOME 150
      WRITE (6,160) (J3,I,A(I,NBRXYW),I=1,NBRX)                         SOME 160
      WRITE (6,170) A(NOVAR,NBRXYW)                                     SOME 170
      WRITE (6,180)                                                     SOME 180
      WRITE (6,190) ((JB,I,J,A(I,J),J=I,NBRX),I=1,NBRX)                 SOME 190
      WRITE (6,200) (J9,I,A(I,NOVAR),I=1,NBRX)                          SOME 200
      WRITE (6,210) A(NOVAR,NOVAR)                                      SOME 210
      RETURN                                                            SOME 220
C     ******************************************************************SOME 230
C     ******************************************************************SOME 240
C     S/R  TO CALCULATE AND PRINT THE RESIOUAL SUM OF SQUARES AND C.F.  SOME 250
      ENTRY RESIO                                                       SOME 260
      IF (A(NBRXYW,NBRXYW)) 10,10,20                                    SOME 270
   10 WRITE (6,220) NPROB                                               SOME 280
      CALL EXIT                                                         SOME 290
   20 OO 40 I=1,NOVAR                                                   SOME 300
      OO 30 J=I,NOVAR                                                   SOME 310
   30 A(I,J)=A(I,J)-(A(I,NBRXYW)*A(J,NBRXYW)/A(NBRXYW,NBRXYW))          SOME 320
   40 AVG(I)=A(I,NBRXYW)/A(NBRXYW,NBRXYW)                               SOME 330
      IF (IFAVE.NE.0) GO TO 50                                          SOME 340
      WRITE (6,230)                                                     SOME 350
      WRITE (6,190) ((JB,I,J,A(I,J),J=I,NBRX),I=1,NBRX)                 SOME 360
      WRITE (6,200) (JB,I,A(I,NOVAR),I=1,NBRX)                          SOME 370
      WRITE (6,210) A(NOVAR,NOVAR)                                      SOME 380
   50 RETURN                                                            SOME 390
C     ******************************************************************SOME 400
C     ******************************************************************SOME 410
C     S/R  TO CALCULATE AND PRINT THE SIMPLE CORRELATION COEFFICIENTS   SOME 420
      ENTRY CORREL                                                      SOME 430
      OO 90 I=1,NOVAR                                                   SOME 440
      IF (A(I,I)) 60,60,80                                              SOME 450
   60 WRITE (6,240) I                                                   SOME 460
      IF( I.EQ.NOVAR )  CALL SLITE(1)                                   SOME 470
      SIGMA(I)=1.0                                                      SOME 480
      OO 70 J=1,NOVAR                                                   SOME 490
      A(I,J)=0.0                                                        SOME 500
   70 A(J,I)=0.0                                                        SOME 510
      GO TO 90                                                          SOME 520
CC 80 SIGMA(I)=OSQRT(A(I,I))                                            SOME 530
   80 SIGMA(I) = SQRT ( A(I,I) )
   90 A(I,I)=1.000                                                      SOME 540
      OO 100 I=1,NBRX                                                   SOME 550
```

```
SUBROUTINE PRTSUM     74/74    OPT=1                          FTN 4.2+74278       11/22/74  18.18.48.

            IP1=I+1                                                               SOME 560
            DO 100 J=IP1,NOVAR                                                    SOME 570
            A(I,J)=A(I,J)/(SIGMA(I)*SIGMA(J))                                     SOME 580
        100 A(J,I)=A(I,J)                                                         SOME 590
            DO 110 J=1,NOVAR                                                      SOME 600
            DO 110 K=1,NOVAR                                                      SOME 610
        110 SIMCOR(J,K)=A(J,K)                                                    SOME 620
            IF (IFCNST.NE.0) GO TO 140                                            SOME 630
            IF (IFCORR) 140,120,140                                               SOME 640
        120 WRITE (6,250)                                                         SOME 650
            IF( NBRX.LE.1 )  GO TO 135                                            SOME 660
            NOVM2=NBRX-1                                                          SOME 670
            DO 130 I=1,NOVM2                                                      SOME 680
            IP1=I+1                                                               SOME 690
        130 WRITE (6,260) (J3,I,J,A(I,J),J=IP1,NBRX)                              SOME 700
        135 WRITE (6,270) (J8,I,A(I,NOVAR),I=1,NBRX)                              SOME 710
        140 RETURN                                                                SOME 720
      C                                                                           SOME 730
      C                                                                           SOME 740
      C                                                                           SOME 750
        150 FORMAT (1H0,40X,18HSUM  OF  VARIABLES/ )                              SOME 760
        160 FORMAT (4(A1,11H      SUM X(I2,3H) =F14.4))                           SOME 770
        170 FORMAT (6X,11HSUN     Y  =F14.4)                                      SOME 780
        180 FORMAT (1H070H                              RAW  SUM  CF  SQUARES  ASOME 790
           1ND  CROSS PRODUCTS/)                                                  SOME 800
        190 FORMAT (3(A1,6H    X(I2,7H) VS X(I2,3H) =F17.6))                      SOME 810
        200 FORMAT (3(A1,6H    X(I2,12H) VS    Y  =F17.6))                        SOME 820
        210 FORMAT (5X,16HY      VS     Y  =F17.6)                                SOME 830
        220 FORMAT (32H0 ZERO NUMBER OF DATA, PROBLEM  I6/)                       SOME 840
        230 FORMAT (1H025X56HSUMS  OF  SQUARES  AND  CROSS PRODUCTS  ABCUT  THSOME 850
           1E  MEAN/)                                                             SOME 860
        240 FORMAT (10H0 VARIABLEI5,13H IS CONSTANT //)                           SOME 870
        250 FORMAT (1H0,33X,33HSIMPLE  CORRELATION  COEFFICIENTS/ )               SOME 880
        260 FORMAT (3(A1,6H    X(I2,7H) VS X(I2,3H) =F12.8,5X))                   SOME 890
        270 FORMAT (3(A1,6H    X(I2,12H) VS    Y  =F12.8,5X))                     SOME 900
            END                                                                   SOME 910
```

```
SUBROUTINE ADDTO        74/74    OPT=1                         FTN 4.2+74270        11/22/74  10.10.51.

              SUBROUTINE ADDTO                                                        ADJO  10
        C     S/R  TO ADD A VARIABLE                                                  ADJO   0
              COMMON SIGMA(60),A(52,52),SIMCOR(52,52),AVG(60),TEST(60)                ADJO  20
              COMMON POINT(60),STRING(3000),INOPAC(30,30),INDEXP(61)                  ADJO  30
              COMMON INDEX(60),NOUT(60),KSTEP(60),ALPHA(16),YMEAN,IDEN,IFAVE          ADJO  40
              COMMON MAXSTP,IFPNCH,NSUMRY,NSKIP,NTAPE9,NEW                            ADJO  50
              COMMON NOVAR,NBRNOW,NOSTEP,NDATA,NBRXYN,NBRX,LPATH,DEFRM,KVAR           ADJO  60
              COMMON IFBACK,IFCNST,IFCORR,NPROB,NBRPVR,TOL,MODEL,JF                   ADJO  70
              COMMON INDEXY,LBAD,NOGOOD(20)                                           ADJO  80
              COMMON IFNT,YCONST,NYTRA,V(2,51),YTRA(2)                                ADJO  90
              COMMON  STDERR(50),CORSQR(50)                                           ADJO 100
              EQUIVALENCE (KVAR,K)
              REAL              A,SIMCOR,SIGMA,AVG,TEST                               ADJO 110
        C                                                                             ADJO 120
              REAL              DA,VAR,VMIN,VMAX                                      ADJO 130
        C         NBRNOW = NUMBER OF COEFFICIENTS FOR PRESENT  EQUATION               ADJO 140
        C         NBRPRV = NUMBER OF COEFFICIENTS FOR PREVIOUS EQUATION               ADJO 150
        C         INDEX  = INDEX OF PRESENT  EQUATION                                 ADJO 160
        C         INDEXP = INDEX OF PREVIOUS EQUATION                                 ADJO 170
              DO 10 J=1,NBRNOW                                                        ADJO 180
           10 CALL PACK (NBRNOW,J,INDEX(J),2)                                         ADJO 190
              DO 20 J=1,NBRX                                                          ADJO 200
           20 NOUT(J)=0                                                               ADJO 210
              IF (NBRNOW) 50,50,30                                                    ADJO 220
           30 DO 40 J=1,NBRNOW                                                        ADJO 230
              NDUMMY=INDEX(J)                                                         ADJO 240
           40 NOUT(NDUMMY)=1                                                          ADJO 250
           50 VMAX=-1.0                                                               ADJO 260
        C         FIND LARGEST DELTA                        (VAR = DELTA)             ADJO 270
              DO 70 I=1,NBRX                                                          ADJO 280
        C                              BYPASS IF ALREADY IN EQUATION                  ADJO 290
              IF (NOUT(I).NE.0) GO TO 70                                              ADJO 300
              IF (A(I,I).GE.TOL) GO TO 60                                             ADJO 310
              WRITE (6,510) A(I,I),I,(INDEX(J),J=1,NBRNOW)                            ADJO 320
              GO TO 70                                                                ADJO 330
        C                                                                             ADJO 340
           60 VAR=A(I,NOVAR)*A(NOVAR,I)/A(I,I)                                        ADJO 350
              IF (VAR.LE.VMAX) GO TO 70                                               ADJO 360
              VMAX=VAR                                                                ADJO 370
              K=I                                                                     ADJO 380
           70 CONTINUE                                                                ADJO 390
        C             HAVE FOUND OPTIMAL VARIABLE                                     ADJO 400
              NSTEP=NOSTEP+1                                                          ADJO 410
              IF (VMAX) 80,90,90                                                      ADJO 420
           80 WRITE (6,520) VMAX                                                      ADJO 430
              CALL SLITE (1)                                                          ADJO 440
              GO TO 260                                                               ADJO 450
           90 NBRPVR=NBRNOW                                                           ADJO 460
              NBRNOW=NBRNOW+1                                                         ADJO 470
              IF (TEST(NBRNOW)-A(NOVAR,NOVAR)+VMAX) 100,100,120                       ADJO 480
          100 WRITE (6,530) K,NBRNOW,NSTEP                                            ADJO 490
              DO 110 I=1,NOVAR                                                        ADJO 500
              DO 110 J=1,NOVAR                                                        ADJO 510
          110 A(I,J)=SIMCOR(I,J)                                                      ADJO 520
              LPATH=2                                                                 ADJO 530
              GO TO 260                                                               ADJO 540
        C         ADD VARIABLE TO INDEX                                               ADJO 550
```

```
  120 CONTINUE                                                    ADJO 560
      IF (NBRPVR) 230,230,130                                     ADJO 570
  130 DO 140 J=1,NBRPVR                                           ADJO 580
  140 CALL PACK (NBRPVR,J,INDEXP(J),2)                            ADJO 590
      DO 150 J=1,NBRNOW                                           ADJO 600
  150 CALL PACK (NBRNOW,J,INDEX(J),2)                             ADJO 610
      DO 180 J=1,NBRPVR                                           ADJO 620
      IF (INDEXP(J)-K) 180,160,170                                ADJO 630
  160 CALL SLITE (1)                                              ADJO 640
      WRITE (6,480)                                               ADJO 650
      GO TO 260                                                   ADJO 660
  170 JJ=J                                                        ADJO 670
      GO TO 190                                                   ADJO 680
  180 CONTINUE                                                    ADJO 690
      INDEXP(NBRNOW)=K                                            ADJO 700
      GO TO 210                                                   ADJO 710
  190 L=NBRNOW-JJ                                                 ADJO 720
      DO 200 J=1,L                                                ADJO 730
      NR=NBRNOW+1-J                                               ADJO 740
      NS=NBRNOW-J                                                 ADJO 750
  200 INDEXP(NR)=INDEXP(NS)                                       ADJO 760
      INDEXP(JJ)=K                                                ADJO 770
C          CHECK TO SEE IF SET HAS ALREADY BEEN COMPUTED          ADJO 780
  210 DO 220 J=1,NBRNOW                                           ADJO 790
      CALL PACK (NBRNOW,J,I,2)                                    ADJO 800
      IF (INDEXP(J).NE.I) GO TO 240                               ADJO 810
  220 CONTINUE                                                    ADJO 820
      WRITE (6,490) NOSTEP,K,NBRNCH,(INDEX(J),J=1,NBRNCH)         ADJO 830
      LPATH=3                                                     ADJO 840
      GO TO 260                                                   ADJO 850
C                                                                 ADJO 860
  230 INDEXP(1)=K                                                 ADJO 870
  240 TEST(NBRNCH)=A(NOVAR,NOVAR)-VMAX                            ADJO 880
C          NEW SET - PUT INDEXES IN MATRIX                        ADJO 890
      DO 250 J=1,NBRNOW                                           ADJO 900
  250 CALL PACK (NBRNOW,J,INDEXP(J),1)                            ADJO 910
      LPATH=1                                                     ADJO 920
      IF(MAXSTP.EQ.999) GO TO 255                                 ADJO 930
      IF(NSTEP.EQ.1) GO TO 256                                    ADJO 940
  255 WRITE(6,500) NSTEP,K, V(1,K),V(2,K)                         ADJO 950
  256 KSTEP(NBRNCH)=NSTEP                                         ADJO 960
  260 RETURN                                                      ADJO 980
C     *************************************************************ADJO 990
C     *************************************************************ADJO1000
C                                                                 ADJO1010
C          ADJUST CORRELATION MATRIX FOR ENTRANCE OF VARIABLE K   ADJO1020
      ENTRY MATRIX                                                ADJO1030
C                                                                 ADJO1040
      DA=1.0/A(K,K)                                               ADJO1050
      DO 300 I=1,NOVAR                                            ADJO1060
      IF (I-K) 270,300,270                                        ADJO1070
  270 DO 290 J=1,NOVAR                                            ADJO1080
      IF (J-K) 280,290,280                                        ADJO1090
  280 A(I,J)=A(I,J)-A(I,K)*A(K,J)*DA                              ADJO1100
  290 CONTINUE                                                    ADJO1110
      A(I,K)=-A(I,K)*DA                                           ADJO1120
  300 CONTINUE                                                    ADJO1130
```

```
      SUBROUTINE ADDTO      74/74    OPT=1                    FTN 4.2+74278       11/22/74  18.18.52.

            DO 320 J=1,NOVAR                                                      ADJ01140
            IF (J-K) 310,320,310                                                  ADJ01150
        310 A(K,J)=A(K,J)*OA                                                      ADJ01160
        320 CONTINUE                                                              ADJ01170
            A(K,K)=OA                                                             ADJ01180
      C                                                                           ADJ01190
            IF (NOSTEP.LE.1) NCOMB=MIN1(1000.,2.**(NOVAR-1))                      ADJ01200
            IF (NOSTEP.LE.NCOMB) GO TO 330                                        ADJ01210
            WRITE (6,540) NPROO                                                   ADJ01220
            CALL SLITE (1)                                                        ADJ01230
        330 RETURN                                                                ADJ01240
      C     **********************************************************************ADJ01260
      C     S/R  TO DELETE A VARIAOLE                                             ADJ01270
            ENTRY REMOVE                                                          ADJ01280
      C                                                                           ADJ01290
            NODEL = 0
            IF( NORNOW.LE.1 )  GO TO 470
            DO 340 J=1,NBRNOW                                                     ADJ01300
        340 CALL PACK (NBRNOW,J,INDEX(J),2)                                       ADJ01310
            K=0                                                                   ADJ01330
            NSTEP=NOSTEP+1                                                        ADJ01340
            VMIN=-2.0E+30                                                         ADJ01350
      C          FIND SMALLEST DELTA                              (VAR = DELTA)   ADJ01360
            DO 360 J=1,NBRNOW                                                     ADJ01370
            I=INDEX(J)                                                            ADJ01380
            IF (A(I,I)-TOL) 360,360,350                                           ADJ01390
        350 VAR=A(I,NOVAR)*A(NOVAR,I)/A(I,I)                                      ADJ01400
            IF (VAR.LE.VMIN) GO TO 360                                            ADJ01410
            VMIN=VAR                                                              ADJ01420
            K=I                                                                   ADJ01430
        360 CONTINUE                                                              ADJ01440
            IF (TEST(NBRNOW-1)+VMIN-A(NOVAR,NOVAR)) 370,370,380                   ADJ01450
        370 CONTINUE                                                              ADJ01460
            GO TO 471                                                             ADJ01470
      C          REMOVE VARIABLE TO BE DELETED FROM INDEX                         ADJ01480
        380 JP = 0                                                                ADJ01490
            DO 390 J=1,NBRNOW                                                     ADJ01500
            IF( INDEX(J).EQ.K ) GO TO 390                                         ADJ01510
            JP = JP + 1                                                           ADJ01520
            INDEXP(JP) = INDEX(J)                                                 ADJ01530
        390 CONTINUE                                                              ADJ01540
            NBRPVR = NBRNOW - 1                                                   ADJ01550
      C          CHECK TO SEE IF SET HAS ALREADY BEEN COMPUTED                    ADJ01560
            DO 440 J=1,NBRPVR                                                     ADJ01570
            CALL PACK (NBRPVR,J,I,2)                                              ADJ01580
            IF (I.NE.INDEXP(J)) GO TO 450                                         ADJ01590
        440 CONTINUE                                                              ADJ01600
            WRITE (6,550) K                                                       ADJ01610
            GO TO 470                                                             ADJ01620
      C          NEW SET - PUT INDEXES IN MATRIX                                  ADJ01630
        450 NODEL = 1                                                             ADJ01640
            NBRNOW = NBRNOW - 1                                                   ADJ01650
            TEST(NBRNOW)=-VMIN+A(NOVAR,NOVAR)                                     ADJ01660
            DO 460 J=1,NBRNOW                                                     ADJ01670
        460 CALL PACK (NBRNOW,J,INDEXP(J),1)                                      ADJ01680
            WRITE(6,560) NSTEP,K, V(1,K),V(2,K)                                   ADJ01690
            KSTEP(NBRNOW)=NSTEP                                                   ADJ01700
```

```
SUBROUTINE ADDTO      74/74    OPT=1                         FTN 4.2+74278      11/22/74  18.18.52.

            470 RETURN                                                                  ADJ01710
          C                                                                             ADJ01720
          C                                                                             ADJ01730
          C                                                                             ADJ01740
            500 FORMAT ( 8H0AT STEPI4,20H,  ADJOINED VARIABLEI3 , 3X2A8 )               ADJ01750
            560 FORMAT ( 8H0AT STEPI4,20H,  DELETED  VARIABLEI3 , 3X2A8 )               ADJ01760
            490 FORMAT ( 8H0AT STEPI4,20H,  ADJUNCTION OF VARI3,25H PRODUCES THE SADJ01770
               *AME SET OFI3,21H VARIABLES AS BEFORE ,16I3/42I3)                        ADJ01780
            550 FORMAT ( 1H0,11X      ,20H   DELETION   OF VARI3,45H PRODUCES THE SADJ01790
               *AME SET OF VARIABLES AS BEFORE)                                         ADJ01800
            480 FORMAT (42H0ERROR IN ADJOIN S/R... PROBLEM TERMINATED)                  ADJ01810
            530 FORMAT (21H0 ADJOINING VARIABLE I4,30H  PRODUCES NO IMPROVEMENT FOADJ01820
               1R I3,21H  VARIABLES, AT STEP I4)                                        ADJ01830
            510 FORMAT (39H0NORMALIZED RESIDUAL SUM OF SQUARES ISF11.8,17H .......ADJ01840
               1.VARIABLEI3,31H IS NEAR-DEPENDENT ON VARIABLES11I3,/2X40I3)             ADJ01850
            540 FORMAT (49H0MAXIMUM NUMBER OF ITERATIONS EXCEEDED IN PROBLEMI5)         ADJ01860
            520 FORMAT (14H0NEGATIVE VMAX F10.6, 25H ...... PROBLEM TERMINATED)         ADJ01870
                END                                                                     ADJ01880
```

```
SUBROUTINE PACK       74/74    OPT=1                        FTN 4.2+74278      11/22/74  18.15.12.

          SUBROUTINE PACK (II,JJ,JVALUE,NGO)                                   PACK  20
C              PACK AND EXTRACT INDEX FROM ARRAY NN WHICH HAS 4 VALUES PER WORPACK    0
C                                                                              PACK  10
          COMMON SIGMA(60),A(52,52),SIMCOR(52,52),AVG(60),TEST(60)             PACK  30
          COMMON POINT(60),STRING(3000),INDPAC(30,30),INDEXP(61)               PACK  40
          COMMON INDEX(60),NOUT(60),KSTEP(60),ALPHA(16),YMEAN,IDEN,IFAVE       PACK  50
          COMMON MAXSTP,IFPNCH,NSUMPY,NSKIP,NTAPE9,NEW                         PACK  60
          COMMON NCVAR,NBRNOW,NOSTEP,NDATA,NBRXYN,NBRX,LPATH,DEFRM,KVAR        PACK  70
          COMMON IFBACK,IFCNST,IFCORR,NPROB,NBRPVR,TCL,NCDFL,JF                PACK  80
          COMMON INDEXY,LBAD,NOGOOD(28)                                        PACK  90
          COMMON IFWT,YCCNST,NYTRA,V(2,51),YTRA(2)                             PACK 100
          COMMON   STDERR(53),CORSOR(50)                                       PACK 110
          REAL            A,SIMCOR,SIGMA,AVG,TEST                              PACK 120
          EQUIVALENCE (M,XM)                                                   PACK 130
          DIMENSION MASK(4), MASKC(4), MASKS(4)                                PACK 140
          DATA MASK/77B,7700B,770000B,77000000B/                               PACK 150
          DATA MASKC(1)/77777777777777777700B/
          DATA MASKC(2)/77777777777777770077B/
          DATA MASKC(3)/77777777777777007777B/
          DATA MASKC(4)/77777777777700777777B/
          DATA MASKS/1B,100B,10000B,1000000B/                                  PACK 180
C                                                                              PACK 190
C              INDEX IN NN  ARRAY                                              PACK 200
          I=(II+1)/2                                                           PACK 210
          J=(JJ+1)/2                                                           PACK 220
C              WORD TO BE PACKED                                               PACK 230
          M=INOPAC(I,J)                                                        PACK 240
C              POSITION IN M TO BE PACKED ( 1-4 )                              PACK 250
          NW=2*MOD(II,2)+MOD(JJ,2)+1                                           PACK 260
C                                                                              PACK 270
          GO TO (10,20), NGO                                                   PACK 280
C              PACK INDEX                                                      PACK 290
       10 XM=AND(M,MASKC(NW))                                                  PACK 300
          XM=OR(M,JVALUE*MASKS(NW))                                            PACK 310
          INOPAC(I,J)=M                                                        PACK 320
          RETURN                                                               PACK 330
C              EXTRACT  INDEX                                                  PACK 340
       20 XM=AND(M,MASK(NW))                                                   PACK 350
          JVALUE=M/MASKS(NW)                                                   PACK 360
          RETURN                                                               PACK 370
C                                                                              PACK 380
          END                                                                  PACK 390
```

```
      SUBROUTINE CHANGE (PCINT,NBRTRA,CONST,NTOTAL)                   CHAN  10
C     TRANSFORMATION OF DATA                                           CHAN   0
C                                                                      CHAN  20
      DIMENSION POINT(60), NBRTRA(60), CONST(60)                       CHAN  30
C                                                                      CHAN  40
      DO 220 I=1,NTOTAL                                                CHAN  50
      ITRA=NBRTRA(I)                                                   CHAN  60
      IF (ITRA.LE.0) GO TO 220                                         CHAN  70
      IF (ITRA.LT.17) GO TO 10                                         CHAN  80
    5 WRITE (6,230) ITRA                                               CHAN  90
      NBRTRA(I)=0                                                      CHAN 100
      GO TO 220                                                        CHAN 110
C                                                                      CHAN 120
   10 C=PCINT(I)                                                       CHAN 130
      C=CONST(I)                                                       CHAN 140
      IF (ITRA.GT.8) GO TO 120                                         CHAN 150
      GO TO (20,30,40,50,60,80,100,110), ITRA                          CHAN 160
   20 D=D+C                                                            CHAN 170
      GO TO 210                                                        CHAN 180
   30 C=D*C                                                            CHAN 190
      GO TO 210                                                        CHAN 200
   40 C=D/C                                                            CHAN 210
      GO TO 210                                                        CHAN 220
   50 C=C/C                                                            CHAN 230
      GO TO 210                                                        CHAN 240
   60 IF (C.LT.0.0) GO TO 70                                           CHAN 250
      C=D**C                                                           CHAN 260
      GO TO 210                                                        CHAN 270
   70 C=1./D**(-C)                                                     CHAN 280
      GO TO 210                                                        CHAN 290
   80 IF (D.LT.0.0) GO TO 90                                           CHAN 300
      C=C**D                                                           CHAN 310
      GO TO 210                                                        CHAN 320
   90 D=1./C**(-D)                                                     CHAN 330
      GO TO 210                                                        CHAN 340
  100 D=ALOG(D+C)                                                      CHAN 350
      GO TO 210                                                        CHAN 360
  110 D=ALOG10(D+C)                                                    CHAN 370
      GO TO 210                                                        CHAN 380
C                                                                      CHAN 390
  120 ITRA=ITRA-8                                                      CHAN 400
      IF (C.EQ.0.0) C=1.                                               CHAN 410
      GO TO (130,140,150,160,170,180,190,200), ITRA                    CHAN 420
  130 C=EXP(C*C)                                                       CHAN 430
      GO TO 210                                                        CHAN 440
  140 C=EXP(C/D)                                                       CHAN 450
      GO TO 210                                                        CHAN 460
  150 D=SIN(C*D)                                                       CHAN 470
      GO TO 210                                                        CHAN 480
  160 C=COS(C*D)                                                       CHAN 490
      GO TO 210                                                        CHAN 500
  170 D=SIN(C*C)/COS(C*D)                                              CHAN 510
      GO TO 210                                                        CHAN 520
CC180 C=SINH(C*D)                                                      CHAN 530
CC    GO TO 210                                                        CHAN 540
  180 GO TO 5
CC190 C=COSH(C*D)                                                      CHAN 550
```

```
SUBROUTINE CHARGE      74/74    OPT=1                          FTN 4.2+74278       11/22/74  10.28.55.

CC     GO TO 210                                                               CHAN 560
   190 GO TO 5                                                                 CHAN 570
   200 D=TANH(C*D)                                                             CHAN 580
   210 POINT(I)=D                                                              CHAN 590
   220 CONTINUE                                                                CHAN 600
       RETURN                                                                  CHAN 610
C                                                                              CHAN 620
C                                                                              CHAN 630
C
   230 FORMAT (/1X,14HTRANSFORMATION,I3,18H IS NOT IN TABLES,36H IT WILL BCHAN 640
      1E SET TO ZERO AND IGNORED.//)                                           CHAN 650
       END                                                                     CHAN 660
```

```
SUBROUTINE NBROFX      74/74    OPT=1                        FTN 4.2+74270       11/22/74  10.15.01.

              SUBROUTINE  NBROFX ( IFSUB, POINT )
      C           DUMMY SUBROUTINE TO MAKE UNUSUAL TYPES OF TRANSFORMATIONS.     SRTR   0
      C           THIS SUBROUTINE IS REPLACED AT OBJECT TIME WITH ONE WHICH      SRTR  10
      C           PERFORMS THE DESIRED TRANSFORMATIONS.                          SRTR  20
      C           POINT= DATA OBSERVATION WHICH WAS READ IN ON CARDS OR TAPE     SRTR  30
      C           IFSUB = A NUMBER GREATER THAN ZERO WHICH CALLS S/R EQUAT AND   SRTR  40
      C                 NBROFX.  MAY BE USED AS A BRANCH INDICATOR.               SRTR  50
      C           NXV = NEW NUMBER OF INDEPENDENT VARIABLES                       SRTR  60
      C           MAKE SURE DEPENDENT VARIABLE MATCHES  NXV + INDEXY.             SRTR  70
      C                                                                           SRTR  80
      C           S/R EQUAT IS USED TO ADD OR CHANGE DATA OBSERVATIONS            SRTR  90
      C                                                                           SRTR 100
      C           S/R  NBROFX   IS USED TO CHANGE THE VALUE OF NXV.              SRTR 110
      C                                            AND MUST BE USED.              SRTR 120
      C       **************************************************************SRTR 130
      C                                                                           SRTR 140
      C           EXAMPLE  -  POLYNOMIAL EQUATION      IFSUB = POWER OF X         SRTR 150
      C             Y = A0 + A1*X + A2*X**2 + A3*X**3 + ... + A(IFSUB)*X**IFSUB SRTR 160
      C                                                                           SRTR 170
              DIMENSION POINT(52)                                                 SRTR 250
              EQUIVALENCE ( XV,NXV )
              PRINT 901, IFSUB
          901 FORMAT( "0SUBROUTINE EQUAT WAS CALLED WITH IFSUB =" I3)
              NXV = IFSUB
              POINT(1) = XV

              NY=NXV+1                                                            SRTR 200
              NX=NXV                                                              SRTR 210
              RETURN                                                              SRTR 220
      C                                                                           SRTR 230
              ENTRY EQUAT
      C           STORE Y                                                         SRTR 260
              K=54-NX                                                             SRTR 270
              L=53                                                                SRTR 280
              DO 10 J=NY,52                                                       SRTR 290
              K=K-1                                                               SRTR 300
              L=L-1                                                               SRTR 310
           10 POINT(L)=POINT(K)                                                   SRTR 320
      C                                                                           SRTR 330
              IF( NX.EQ.1 ) RETURN
      C           STORE POWERS OF X                                               SRTR 340
              DO 20 J=2,NX                                                        SRTR 350
           20 POINT(J)=POINT(J-1)*POINT(1)                                        SRTR 360
              RETURN                                                              SRTR 370
              END                                                                 SRTR 380
```

```
      PROGRAM DTANAL(TAPE1=130,TAPE2=65,TAPE11,
     *    PUNCH=65, INPUT=65, OUTPUT=130, TAPE7=PUNCH  )
C
      DIMENSION  TEMP(3), NC(20), ATEMP(3), V(51), NCARD(8)
      DIMENSION  IND(51),COEFF(51),A(855),X(101),Y(101)
      DIMENSION  HEAD(8), LEFT(8), BOTTOM(8), NSYN(5), DROP(5),DL(5)
      DIMENSION  RIGHT(15)
      DATA  DL/8H DH = 12,8H DH = 18,8H DH = 24, 8H DH = 30,8H DH = 21/
      DATA NSYN/1H1,1H2,1H3,1H4,1H5/,DROP/12.,18.,24.,30.,21./
      DATA  TEMP/-65.,70.,160./, NC/20*0/
      DATA ATEMP/10H-65 DEGREE,10H 70 DEGREE,10H160 DEGREE/
      DATA HEAD/8*1H /, RIGHT/15*1H /
      DATA  TP,DH,TC,SS,GL/5*1./
C
      M = 6
      N = 0
C
      NVR = 18
      NVR = 27
      NVR = 48
      NVR = 45
      NVR = 36
C INSERT NUMBER  OF EQUATION TERMS HERE ..................................
C
      REWIND 7
      READ(7,904)  NTEST
      IF( E(F(7) .EQ. 0.0 )  GO TO 200
C
C************* PASS 1 ****************************************************
      NRETURN = 1
      CALL DATE( DAY )
      NC(3)=NC(4)=NC(6)=NC(12)=NC(17)=NC(18)=NC(19)=NC(20)  = 1
      NC(5)=2760
      NC(7)=6
      NC(9)=930
      NC(12) = 10
      PRINT 900, N, M, DAY
  900 FORMAT(*1DYNAMIC CUSHIONING ANALYSIS*10X*N =*I2,10X*N =*I2,9X,A10)
      DO  150  NT = 1,1                                          ********
      REWIND 1
      NP = 0
      NC(1)=NC(20)=NT
      NC(2)=NC(13)=NC(14)=   NVR
      TT = TEMP(NT)
C
  100 READ(1,901)  TP, DH, TC, SS, GL
  901 FORMAT( 10X, 4F10.0, I1, F10.0     CYCLE 1
C 901 FORMAT( 7F10.0 )                   CYCLE 2
      IF( EOF(1) .NE. 0.0 )  GO TO 120
      IF( DH .EQ. 21. )  GO TO 100                              DH=21
      IF( TC .GE. 4. )  GO TO 100
C INSERT DATA CONDITIONS HERE... ALSO IN AFFECTED DO LOOPS IN PASS 2 ..........
C
      YVAR = GL
C
C        COMPUTE DYNAMIC CUSHIONING MODEL VARIABLES
      GO TO 800
```

```
  105 CONTINUE
C
C INSERT YVAR TRANSFORMATION HERE ..............................................
      WRITE(11)   ( V(J), J=1, NVR ),  YVAR
      NP = NP + 1
      GO TO 100
C
  120 END FILE 11
      PRINT 902,NP
  902 FORMAT(//*  NP = *I5)
      WRITE(2,904)  NC
      PRINT   904,  NC
  904 FORMAT( 16I5 )
      WRITE(2,906)
      PRINT   906
  906 FORMAT(80X,T2,*DYNAMIC CUSHIONING MODEL* )
  150 CONTINUE
      REWIND 11
      REWIND 2
      GO TO 999
C
C************* PASS 2 ***************************************************************
C
  200 REWIND 7
      NRETURN = 2
      XMIN = ALOG10( .04 )
      XMAX = ALOG10(  5. )
      DX = ( XMAX - XMIN ) / 100.
  210 READ(7,920)  NCARD,  K, CONST, NT, NV,ND, NSTEP, NO, SE, R
  920 FORMAT(8A10,T1, I5 ,E20.4,5I5,F20.4,F10.7)
      IF( EOF(7) .NE. 0.0 )  GO TO 999
C
CC    PRINT 922, ATEMP(NT), NV, R
      PRINT 922,             NV, R
CC922 FORMAT(*1*A4* DEGREES*I5* VARIABLES    CORRELATION COEFFICIENT I3*
  922 FORMAT(*1*             I5* VARIABLES    CORRELATION COEFFICIENT I3*
     *F6.6///* MLR OUTPUT CARDS*//)
      PRINT 910, NCARD
  910 FORMAT(1X,8A10)
      DO 220 J=1,NV
      READ(7,920)  NCARD, IND(J), COEF(J)
  220 PRINT 910, NCARD
      ENCODE(150,911,RIGHT)  ( IND(J), J=1,NV )
  911 FORMAT(100X,50X,T1, 50I3)
C
      DO 260 NT = 1,3                                               ********
      HEAD(1) = ATEMP(NT)
      TP = TEMP(NT)
C      DROP HEIGHT
      DO 250 ND = 1,5
      DH = DROP(ND)
      IF( DH .EQ. 21. )  GO TO 250                                  D4=21
C**
      HEAD(6) = DL(ND)
      CALL SETGRID( A, =76, XMIN, XMAX, 0.0, 350.     )
      CALL LABGRID( A, 1, 20, 20HLOG( STATIC STRESS ) )
      CALL LABGRID( A, 2, 20, 20H              G LEVEL )
```

```
      PROGRAM DIANAL     74/74   OPT=1                    FTN 4.2+74278      11/21/74  10.51.10.

              CALL LABGRID( A, 3, 48, HEAD )
              CALL LABGRID( A, 4, 76, RIGHT )
        C
        C          THICKNESS
              DO 240 NTC =1,5
              TC = NTC
              IF( TC .GE. 4. )  GO TO 240

              XX = XMIN
              DO 230 JP=1,101
              SS = 10**XX
        C          COMPUTE DYNAMIC CUSHIONING MODEL VARIABLES
              GO TO 800
          222 CONTINUE
              Y(JP) = CONST
              DO 225 J=1,NV
              I = IND(J)
          225 Y(JP) = Y(JP) + COEFF(J) * V(I)
              X(JP) = XX
        C
        C INSERT REVERSE YVAR TRANSFORMATION HERE.    Y(JP) = ----  ......................
        C
          230 XX = XX + DX
        C
              CALL PLTGRID( A, NSYM(NTC), 101, X, Y )
          240 CONTINUE
        C
              CALL PRINTPL( A, 6LOUTPUT )
          250 CONTINUE
          260 CONTINUE                                                           ********
              GO TO 110
        C
          800 CONTINUE
        C********** DYNAMIC CUSHIONING MODEL ********************************************

              SS = SS * 100.
              AL = ALOG( SS )
              AL2 = AL * AL
              SRDM = SQRT( DM )
              TCSM = TC ** (-3.5)
               TR = (TP+460)/100.
              TR2 = TR * TR
              TR3 = TR * TR2
              TR4 = TR3 * TR
        C
              TCOM = TC ** (-0.5)
              TCTM = TC ** (-1.5)
              TCINV= TC ** (-2.5)
        C
              V(61) = TR     * TCOM    * 1.0
              V(62) = TR     * TCOM    * 1.0     * AL
              V(63) = TR     * TCOM    * 1.0     * AL2
              V(64) = TR     * TCTM    * SRDM
              V(65) = TR     * TCTM    * SRDM    * AL
              V(66) = TR     * TCTM    * SRDM    * AL2
              V(67) = TR     * TCOM    * SRDM
              V(68) = TR     * TCOM    * SRDM    * AL
```

```
      PROGRAM OTANAL      74/74    OPT=1                    FTN 4.2+74278        11/21/74  10.51.18.

          V(9) = TR   * TCOM   * SROM   * AL2

          V(10) = TR2  * TCOM   * 1.0
          V(11) = TR2  * TCOM   * 1.0    * AL
          V(12) = TR2  * TCOM   * 1.0    * AL2
          V(13) = TR2  * TCTM   * SROM
          V(14) = TR2  * TCTM   * SROM   * AL
          V(15) = TR2  * TCTM   * SROM   * AL2
          V(16) = TR2  * TCOM   * SROM
          V(17) = TR2  * TCOM   * SROM   * AL
          V(18) = TR2  * TCTM   * SROM   * AL2

          V(19) = TR3  * TCOM   * 1.0
          V(20) = TR3  * TCOM   * 1.0    * AL
          V(21) = TR3  * TCOM   * 1.0    * AL2
          V(22) = TR3  * TCTM   * SROM
          V(23) = TR3  * TCTM   * SROM   * AL
          V(24) = TR3  * TCTM   * SROM   * AL2
          V(25) = TR3  * TCOM   * SROM
          V(26) = TR3  * TCOM   * SROM   * AL
          V(27) = TR3  * TCTM   * SROM   * AL2

          V(28) = TR   * TCTM
          V(29) = TR   * TCTM            * AL
          V(30) = TR   * TCTM            * AL2
          V(31) = TR2  * TCTM
          V(32) = TR2  * TCTM            * AL
          V(33) = TR2  * TCTM            * AL2
          V(34) = TR3  * TCTM
          V(35) = TR3  * TCTM            * AL
          V(36) = TR3  * TCTM            * AL2
C*********************************************
          GO TO ( 185 , 222 ), NRETURN
C
  999 CONTINUE
      END
```

```
      SUBROUTINE  PRINTPL( P, N1, N2, X, Y, Z )
C     SUBROUTINE PRINTPL
C     DESCRIPTION
C         THIS ROUTINE WAS WRITTEN TO ALLOW MULTIPLE PLOTS ON A SINGLE
C         PAGE OF AN ON-LINE PRINTER. THE PRINT CHARACTER MAY BE
C         SELECTED BY THE USER.
C     INPUT
C         THE INPUT TO THIS ROUTINE IS BY THE FOLLOWING ENTRY POINTS.
C       1 CALL SETGRID( P,JL,XMIN,XMAX,YMIN,YMAX )
C         SETGRID MUST BE FIRST CALL. IT SETS UP AND CLEARS ARRAY P.
C         P = ARRAY TO HOLD PLOT. DIMENSION MUST BE AT LEAST  JL*11+18.
C         JL = NUMBER OF LINES IN PLOT. 51 FOR 6 LINES PER INCH OR
C              76 FOR 8 LINES PER INCH IS RECOMMENDED.
C              IF NEGATIVE, INSIDE GRID LINES WILL NOT BE DRAWN.
C         XMIN,XMAX,YMIN,YMAX = MIN AND MAX  OF  X AND Y VALUES.
C       2 CALL LABGRID( P, L, NC, LABEL )
C         LABGRID IS USED TO LABEL PLOT GRID. CALLED FOR EACH LABEL.
C         P = ARRAY THAT HOLDS THE PLOT.
C         L = 1 FOR BOTTOM LABEL
C         L = 2 FOR LEFT   LABEL
C         L = 3 FOR TOP    LABEL
C         L = 4 FOR RIGHT  LABEL
C         NC = NUMBER OF CHARACTERS IN LABEL.
C         LABEL = ARRAY THAT HOLDS THE LABEL.
C                 NOTE- DO NOT CHANGE UNTIL AFTER PLOT HAS BEEN PRINTED.
C       3 CALL PLTGRID( P, NSYM, NP, X, Y )
C         PLTGRID ENTERS DATA INTO PLOT.
C         P = ARRAY THAT HOLDS THE PLOT.
C         NSYM = PLOT SYMBOL ( EX. NSYM=1H* )
C         NP = NUMBER OF POINTS TO BE PLOTTED
C         X = X DATA ARRAY
C         Y = Y DATA ARRAY
C       4 CALL PRINTPL( P, NFILE )
C         PRINTPL PRINTS GRAPH STORED IN ARRAY P. IT DOES NOT CLEAR P.
C         P = ARRAY THAT HOLDS THE PLOT
C         NFILE = OUTPUT FILE THAT THE PLOT IS PRINTED ON.
C     OUTPUT
C         ON-LINE PRINTER PLOTS
C     REMARKS

      DIMENSION   P(1),  X(1),  Y(1),  JWORK(11), WORK(11)
C         P = ARRAY TO HOLD PLOT
C         X =  XMAX, LABEL, X DATA ARRAY
C         Y =  YMIN,         Y DATA ARRAY
C         Z =  YMAX
C         N1 = JL   , L     , NSYM         , NFILE
C         N2 = XMIN, NM    , NP
      EQUIVALENCE ( XJ,JX,XMIN ), ( WORK,JWORK ), ( MASK,PMASK )
      DATA  NFILE/6LOUTPUT/,PMASK/11111111111111111111B/, BLANK/1H /
      GO TO 15
C       SET UP GRID              P, N1, N2  ,X    , Y    ,Z
C************** CALL SET GRID ( P, JL, XMIN,XMAX, YMIN,YMAX ) *****************
C         P = ARRAY TO HOLD PLOT, ITS SIZE MUST BE AT LEAST JL*11+18.
C         JL = NUMBER OF LINES IN PLOT. 51 FOR 6 LINES PER INCH OR
C              76 FOR 8 LINES PER INCH IS RECOMMENDED.
C              IF NEGATIVE, INSIDE GRID LINES WILL NOT BE DRAWN.
      ENTRY  SET GRID
```

```
SUBROUTINE PRINTPL     74/74    OPT=1                         FTN 4.2+ REL        07/25/74  14.19.05.

            JL = IABS( N1 )
            IF( JL .EQ. 0 ) GO TO 90
            GVV = 10HI                                                          PRIN 330
            GVV = 10H]                                                          PRIN 340
            GHH = 10HI---------                                                 PRIN 350
            GHH = 10H]---------                                                 PRIN 360
            GV = GVV                                                            PRIN 370
            GH = GHH                                                            PRIN 380
            GG = GH                                                             PRIN 390
          1 IF( N1 .GE. 0 )  GO TO 2
            GV = BLANK
            GH = BLANK
            GG = GVV
      C         SET UP CODE THAT SETGRID HAS BEEN CALLED
          2 P(1) = PMASK
      C         NUMBER LINES TO GRAPH
            P(2) = JL
            JJ = 4                                                              PRIN 470
            DO 4 I=1.JL                                                         PRIN 480
            G = GV                                                              PRIN 490
            GB = GVV                                                            PRIN 500
            JJ = JJ + 1                                                         PRIN 510
            IF( JJ.NE.5 ) GO TO 3                                               PRIN 520
            G = GH                                                              PRIN 530
            GB = GG                                                             PRIN 540
            JJ = 0                                                              PRIN 550
          3 DO 4 K = 1,11                                                       PRIN 560
      CCC   P(I.K) = G
            IND = 18 + (I-1)*11 + K                                             ????????
            P(IND) = G
            IF( K.EQ.1 .OR. K.EQ.11 )  P(IND) = GB
            IF( I.EQ.1 .OR. I.EQ.JL )  P(IND) = GHH
          4 CONTINUE
      C         COUNT OF POINTS THAT FELL OUT OF GRID
            P(3) = 0.
      C         XMIN
            JX = N2
            P(4) = XNIN
      C         YNAX
            YMAX = Z
            P(5) = YMAX
            XMAX = X(1)
            YNIN = Y(1)
      C         X AND Y SCALE INCREMENTS
            SX = ( XMAX - XMIN ) / 100.
            SY = ( YMAX - YMIN ) / ( P(2) - 1. )
            OX = 1.5 - XMIN / SX
            OY = 1.5 - YMIN / SY
            P(6) = SX
            P(7) = SY
            P(8) = OX
            P(9) = OY
      C         SET LABLE ADDRESSES = 0
            DO 8 J=10,17
          8 P(J) = 0.
            RETURN
      C                                   P. N1, N2. X
```

```
      C*************** CALL  LAB GRID ( P, L , NC, LABEL ) *****************************
      C          LABGRID PUTS LABELS ON GRID AXIS
      C          NC     = NUMBER OF CHAR. IN LABEL ARRAY = N2
      C          LABEL = LABEL ARRAY                      = X
            ENTRY  LAB GRID
            XJ = P(1)
            CALL = 7HLABGRID
            IF( JX .NE. MASK )  GO TO 90
            LABELA = LOCF( X(1) ) - LOCF( P(1) )  + 1
            J = N1*2 + 8
      C     LABEL ADDRESS IN REFERENCE TO P
            P(J) = LABELA
      C     NUMBER OF CHARACTERS IN LABEL
            P(J+1) = N2
            RETURN
      C                                  P, N1  , N2, X, Y
      C*************** CALL PLT GRID ( P, NSYM, NP, X, Y ) ****************************
      C          PLTGRID ENTERS DATA INTO PLOT GRID P
      C          NSYM = PLOT SYMBOL ( EX. PSYM = 1H* )          = N1
      C          NP   = NUMBER OF POINTS TO PLOT                = N2
            ENTRY  PLT GRID
            CALL = 7HPLTGRID
            XJ = P(1)
            IF( JX .NE. MASK )  GO TO 90
            NP = N2
            NSYM = N1
            IF( NP .LE. 0 )  RETURN
            SX = P(6)
            SY = P(7)
            DX = P(8)
            DY = P(9)
            DO 30 L = 1, NP
            IF( LEGVAR( X(L) ).NE.0 )  GO TO 20                        PRIN1340
            IF( LEGVAR( Y(L) ).NE.0 )  GO TO 20                        PRIN1350
            JL = P(2)
            J = X(L) / SX + DX
            I = Y(L) / SY + DY
            IF( J.LT.1 .OR. J.GT.101 )  GO TO 20                       PRIN1380
            IF( I.LT.1 .OR. I.GT.JL  )  GO TO 20                       PRIN1390
      C          INSERT PLOT CHARACTER                                 PRIN1400
            JWORD = ( J-1 )/10 + 1                                     PRIN1410
            JPOS  = J - ( JWORD-1 )*10                                 PRIN1420
            IND = 18 + ( I-1 )*11 + JWORD
            DECODE(10,905, P(IND) ) (JWORK(J), J=1,10)
            JWORK(JPOS) = NSYM
            ENCODE(10,905, P(IND) ) (JWORK(J), J=1,10)
        905 FORMAT(10A1)
            GO TO 30
         20 P(3) = P(3) + 1.0
         30 CONTINUE
      C
      C
            RETURN
      C                                  P, N1
      C*************** CALL PRINT PL ( P, NFILE ) ***************************************
      C          PRINTPL PRINTS GRAPH STORED IN ARRAY P ON OUTPUT FILE NFILE.
      C          IT DOES NOT CLEAR THE GRID.
```

```
   15 XJ = P(1)
      CALL = 7HPRINTPL
      IF( JX .NE. MASK )  GO TO 90
       NFILE = N1
       IF( N1 .EQ. 0 )  NFILE = 6LOUTPUT
       WRITE(NFILE,912)
  912 FORMAT(*T*)                                                       PRIN1608
C           PRINT TOP LABEL
       L1 = P(14)
       IF( L1 .EQ. 0 )  L1 = LOCF( BLANK ) - LOCF( P(1) )  + 1
      L2 = P(14) + P(15)/10.  - .05
       WRITE(NFILE,906)  ( P(L), L=L1,L2 )
  906  FORMAT(1H1,(30X,10A10))
       WRITE(NFILE,907)
  907  FORMAT(8X *FROM       TO*)
       L1 = P(12)
      L2 = P(13)
       L3 = P(16)
      L4 = P(17)
       YMAX = P(5)
       SY = P(7)
C
       Y1 = YMAX - 0.5*SY
       L = -6
       JL = P(2)
       JLP1 = JL + 1
       DO 50 I=1,JL
      L = L + 6
      IF( L .LT. 60 )  GO TO 35
      L1 = L1 + 1
      L3 = L3 + 1
      L = 0
   35 IO2 = 1H
      IO4 = 1H
      IF( L2 .EQ. 0 )  GO TO 40
      IF( I .GT. L2    )  GO TO 40
      IO2 = SHIFT( P(L1), L )
   40 IF( L4 .EQ. 0 )  GO TO 45
      IF( I .GT. L4    )  GO TO 45
      IO4 = SHIFT( P(L3), L )
C
   45 K = JLP1 - I
      Y2 = Y1 + SY
CCC   IND1= 18 +  (K-1)*11
      IND1= 19 +  (K-1)*11
      IND2= IND1+10
      WRITE(NFILE,908)  IO2,Y1,Y2, ( P(J), J=IND1,IND2 ), IO4
  908 FORMAT(1XA1,1XF10.3,1H-,F10.3, 1X,10A10,A1, 1XA1)                 PRIN1740
      Y1 = Y1 - SY
   50 CONTINUE
C
      SX = P(6)
      SXX = SX * 10.
      WORK(1) = P(4)
      DO 60 J=2,11
   60 WORK(J) = WORK(J-1) + SXX
      WRITE(NFILE,910)  ( WORK(J), J=1,11 )
```

```
      910 FORMAT(13X,11F10.2/)
C           BOTTOM LABEL
          L1 = P(10)
          L2 = P(10) + P(11)/10.  - .05
          IF( L1 .EQ  0 )  GO TO 65
          WRITE(NFILE,915)  ( P(L), L=L1,L2 )
      915 FORMAT(30X,10A10)
       65 L  = P(3)
          IF( L .LE. 0 ) GO TO 67
          WRITE(NFILE,902) L
      902 FORMAT(1X,I7, 20H POINTS OUTSIDE GRID )
          GO TO 68
       67 WRITE(NFILE,915) BLANK
       68 WRITE(NFILE,916) SX, SY
      916 FORMAT(*+*,35X,*DX =*F10.3,*    DY =*F10.3)
          WRITE(NFILE,914)
      914 FORMAT(*S*)                                             PRIN1858
          RETURN
C
       90 L = LOCF( P(1) )
          WRITE(NFILE,920)  CALL, L
      920 FORMAT(*0*A8*---- S/R SETGRID NOT CALLED FOR P ARRAY AT LOC *O6)
          LERR = LERR + 1
          IF( LERR .GT. 100 )  STOP 111
          RETURN
          END
```

Appendix C

MLRD PLOTS OF DYNAMIC CUSHIONING CURVES OF THE UAH BEST FITTING POLYNOMIALS

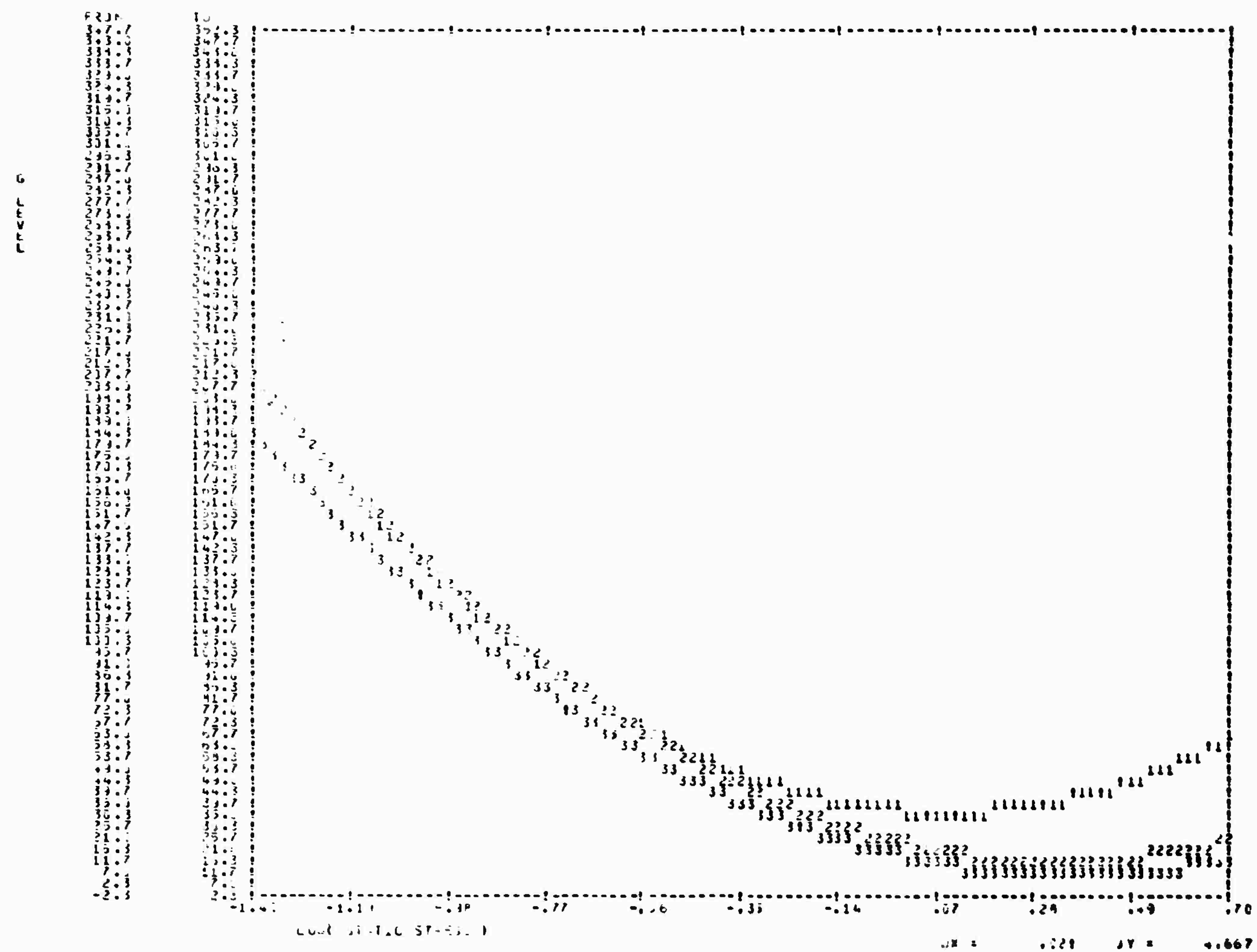


Figure C-1. -65°F, 12-inch drop height.

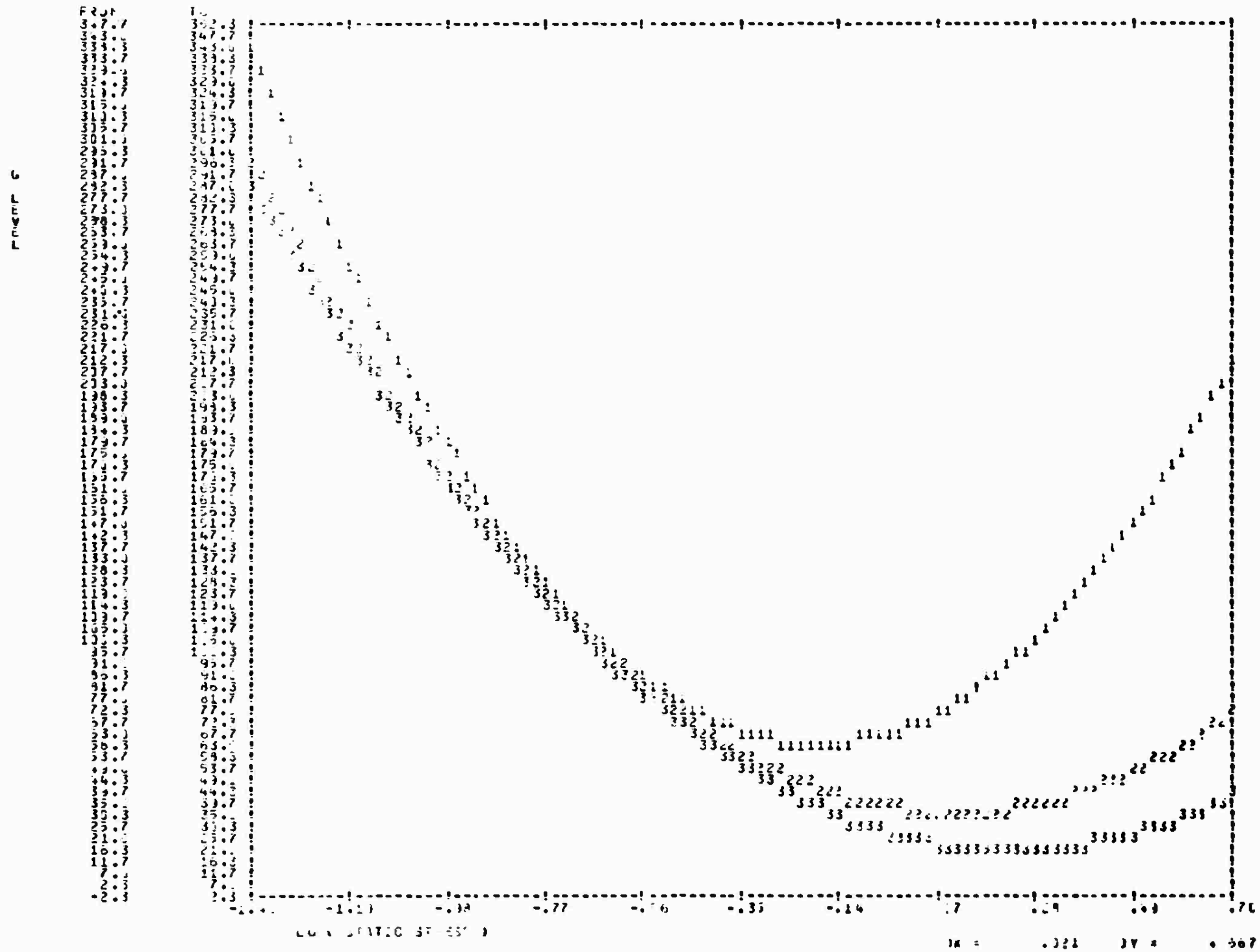


Figure C-2. -65°F, 24-inch drop height.

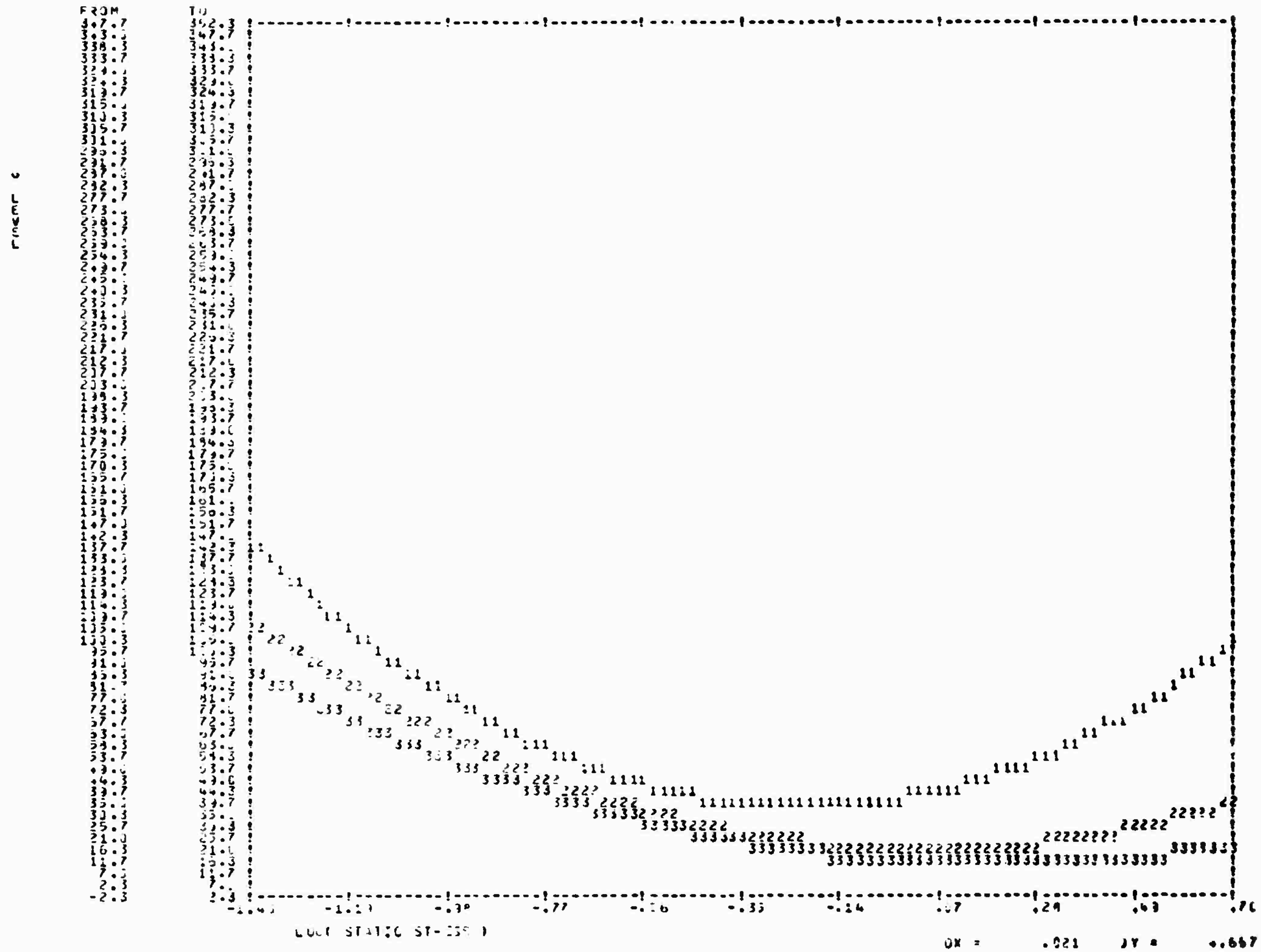


Figure C-3. 70°F, 12-inch drop height.

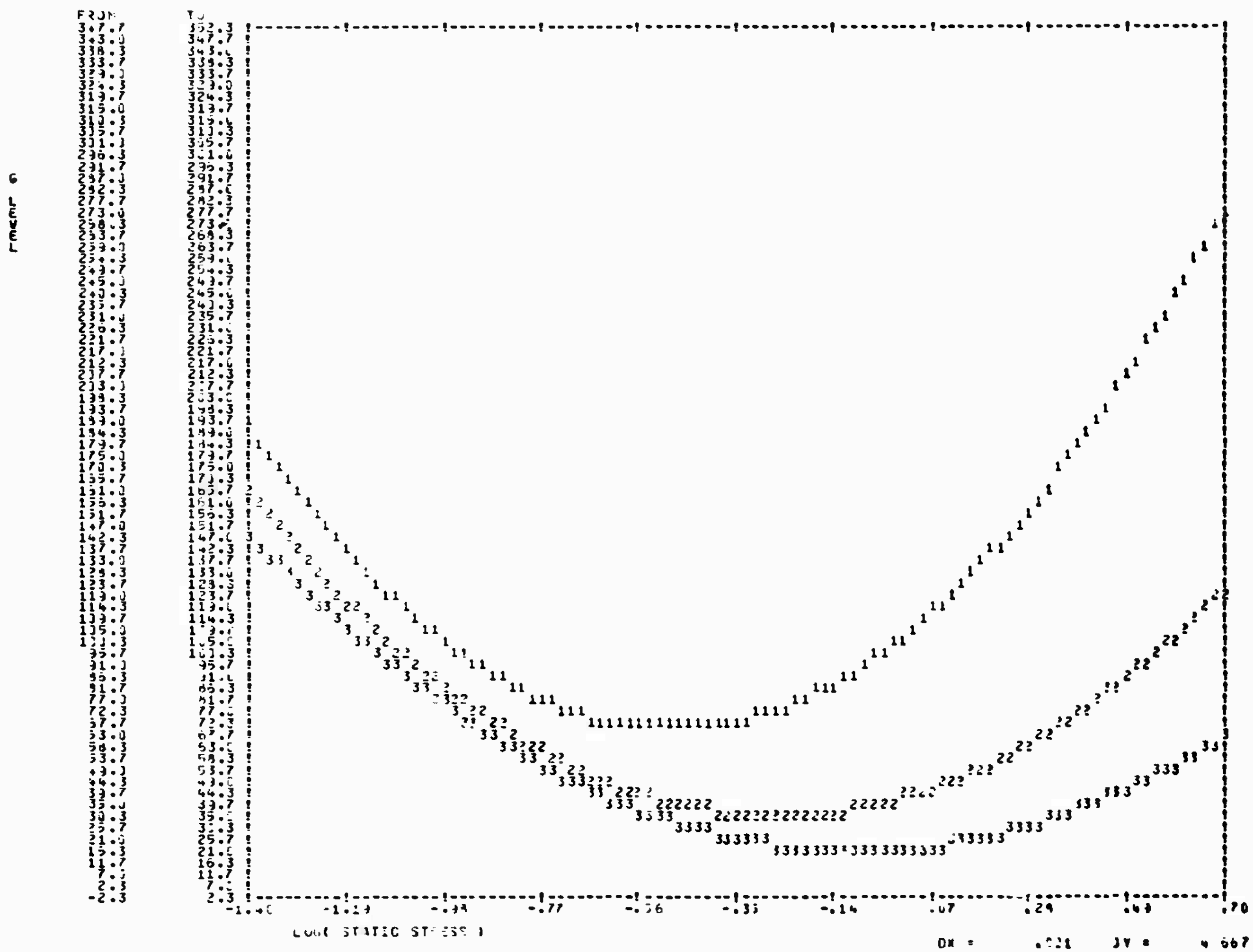


Figure C-4. 70°F, 24-inch drop height.

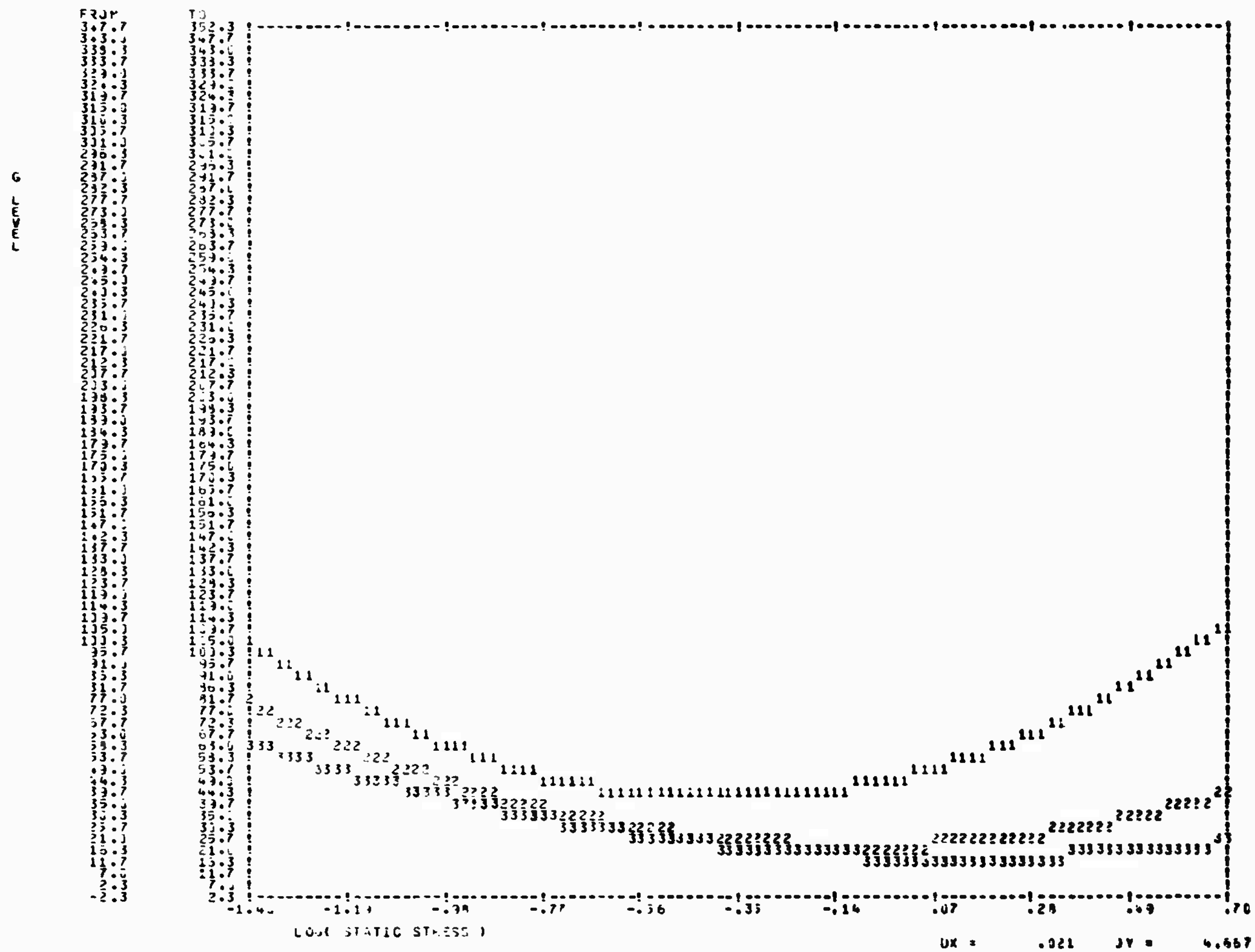


Figure C-5. 160°F, 12-inch drop height.

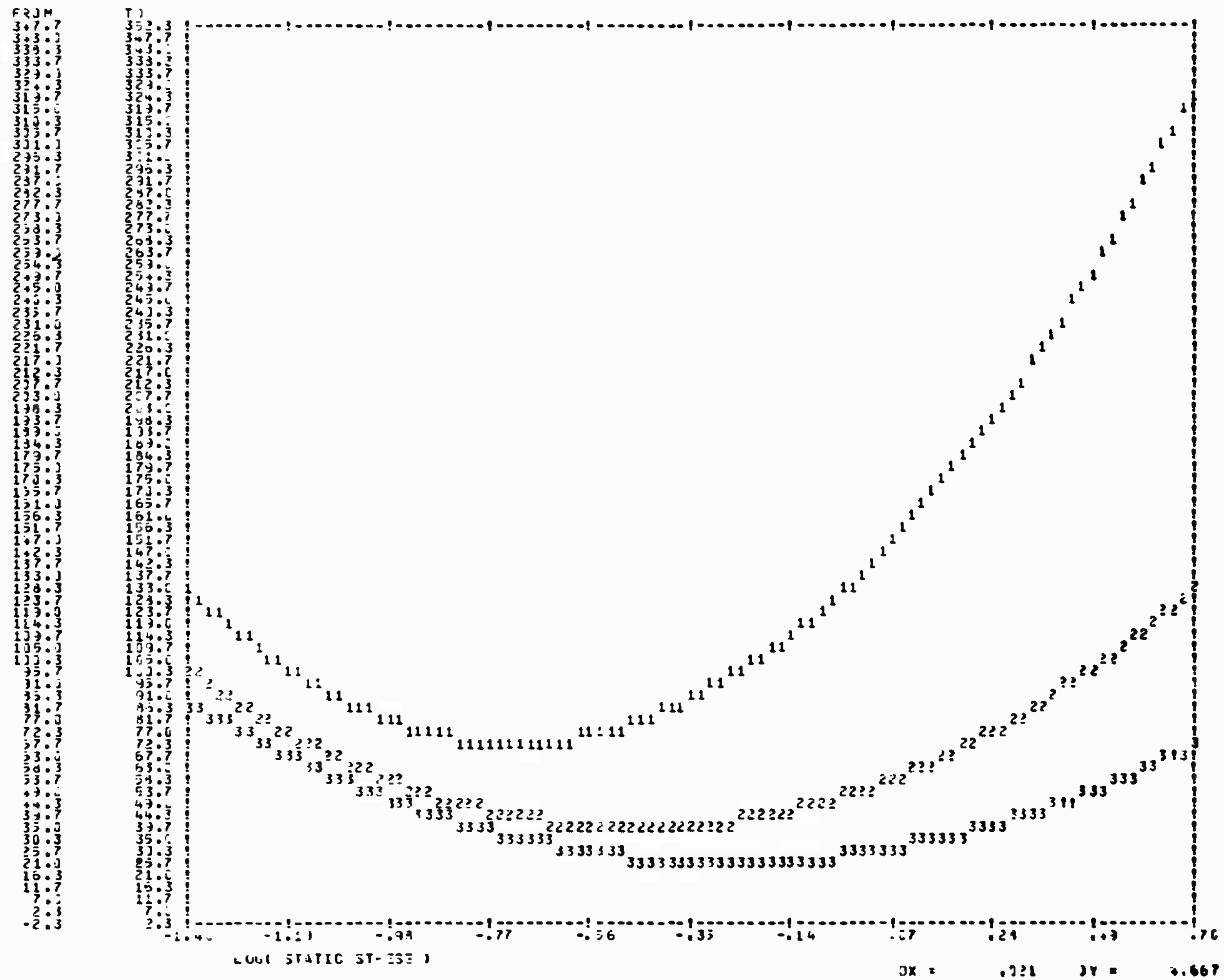


Figure C-6. 160°F, 24-inch drop height.

Appendix D

MLRD PLOTS OF DYNAMIC CUSHIONING CURVES OF THE GENERAL MODEL

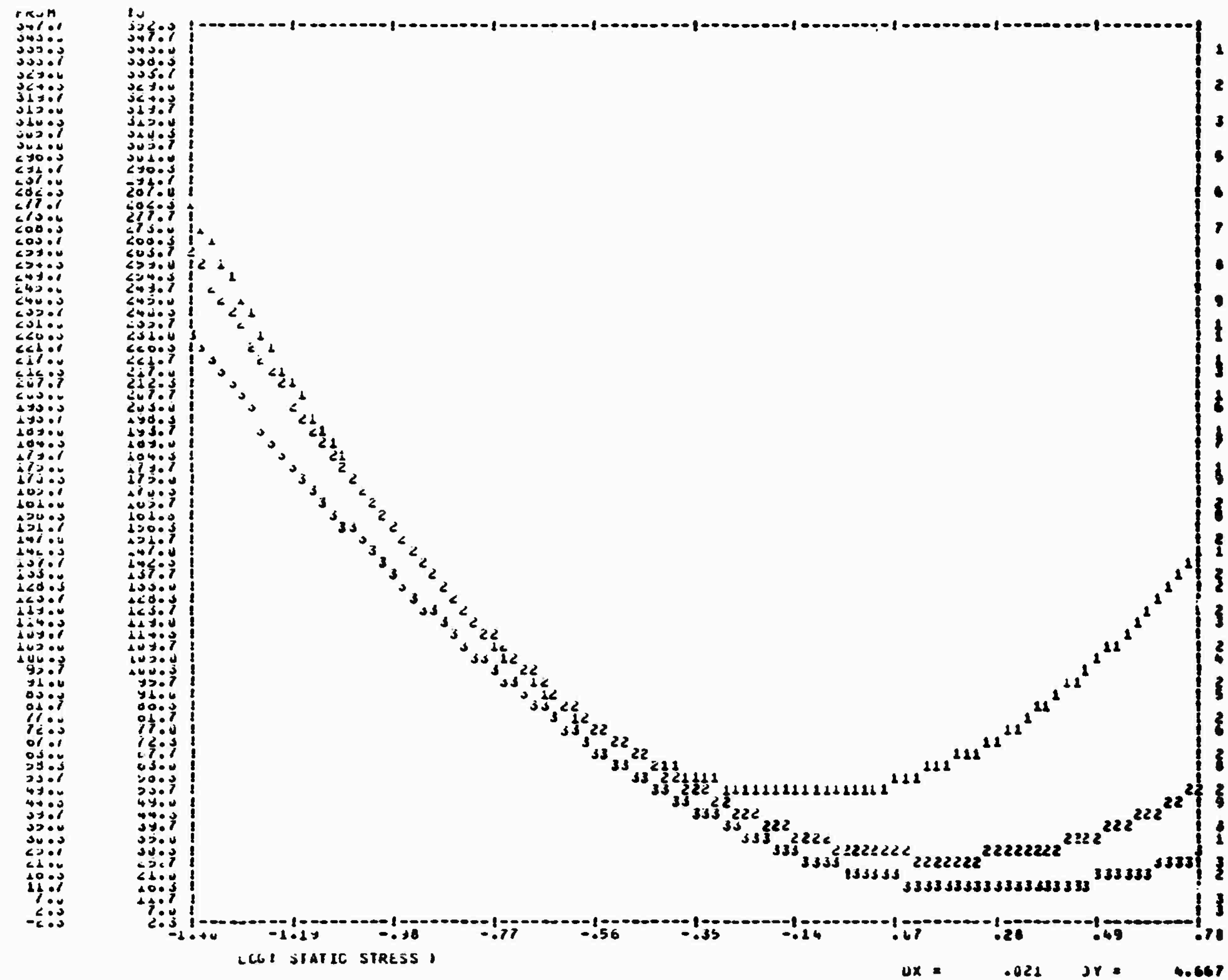


Figure D-1. -65°F, 18-inch drop height.

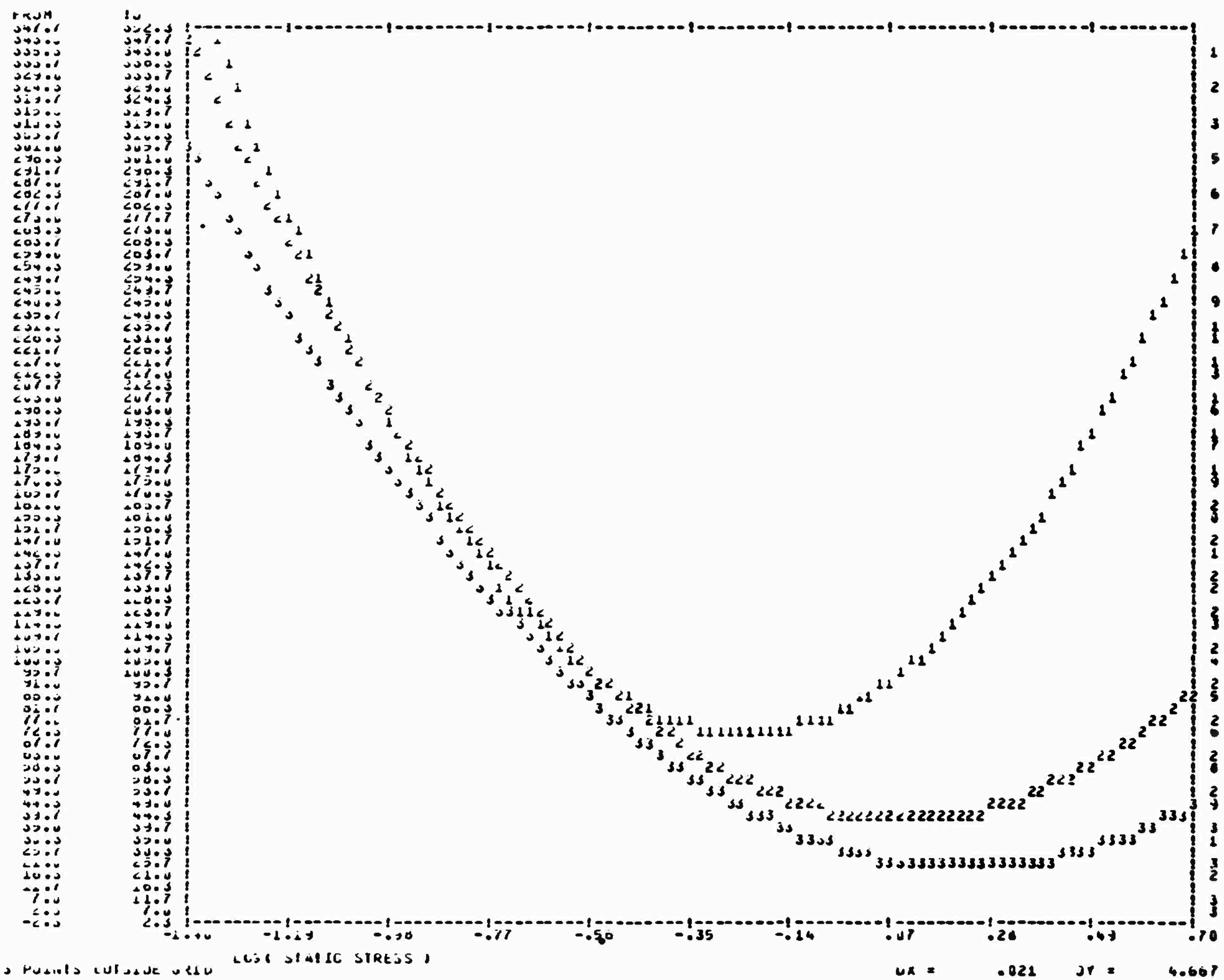


Figure D-2. -65°F, 30-inch drop height.

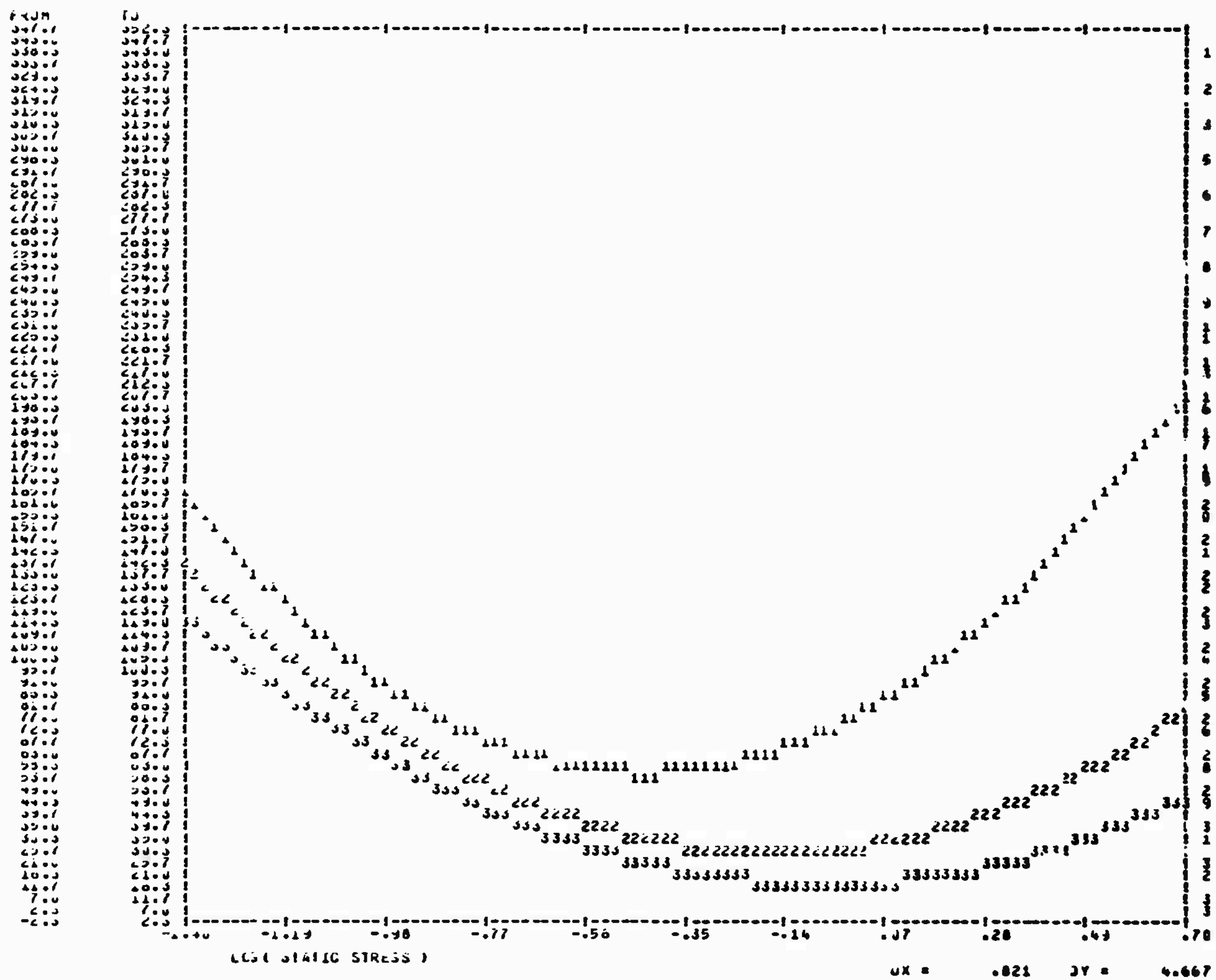


Figure D-3. 70°F, 18-inch drop height.

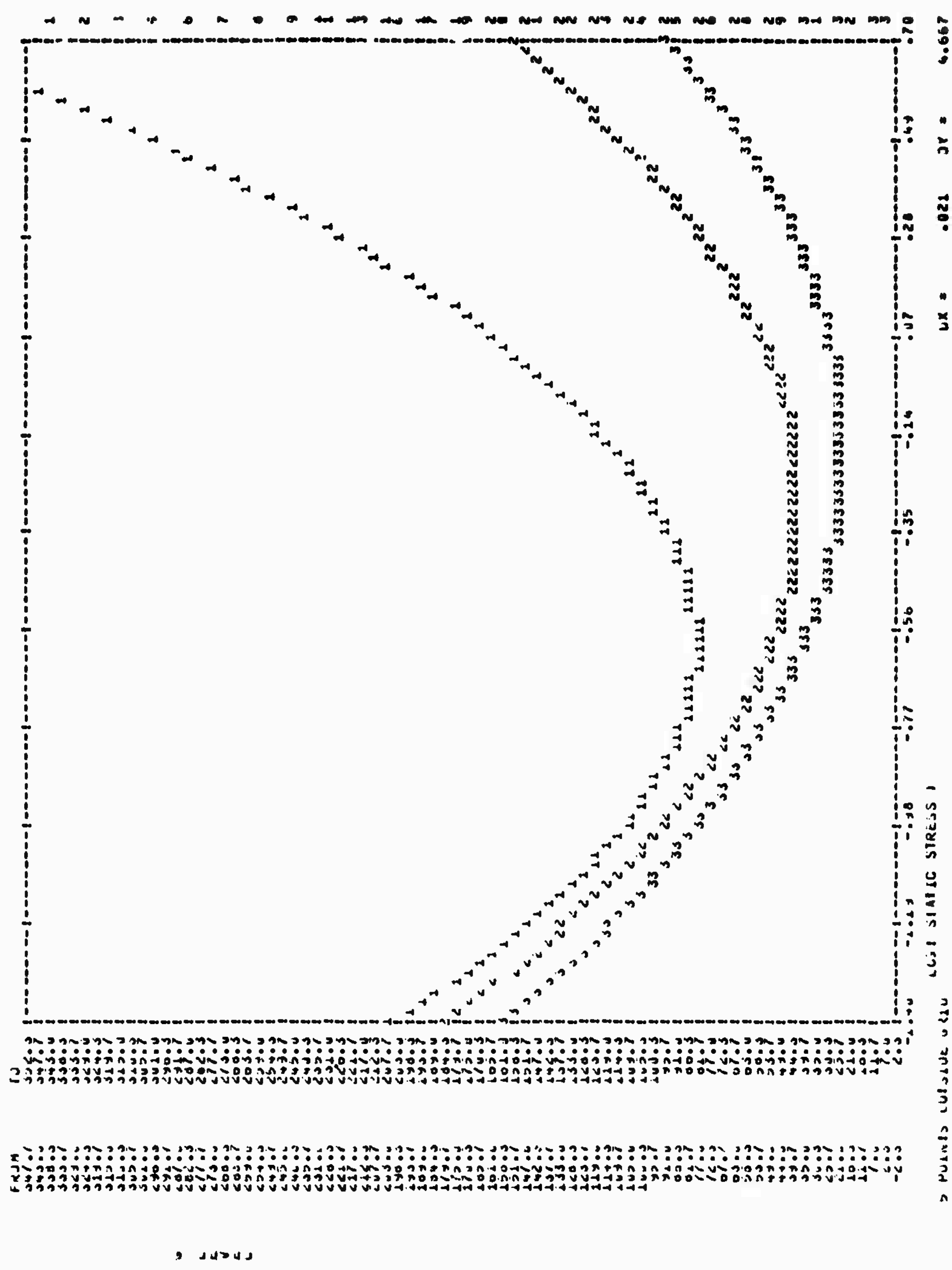


Figure D-4. 70°F, 30-inch drop height.

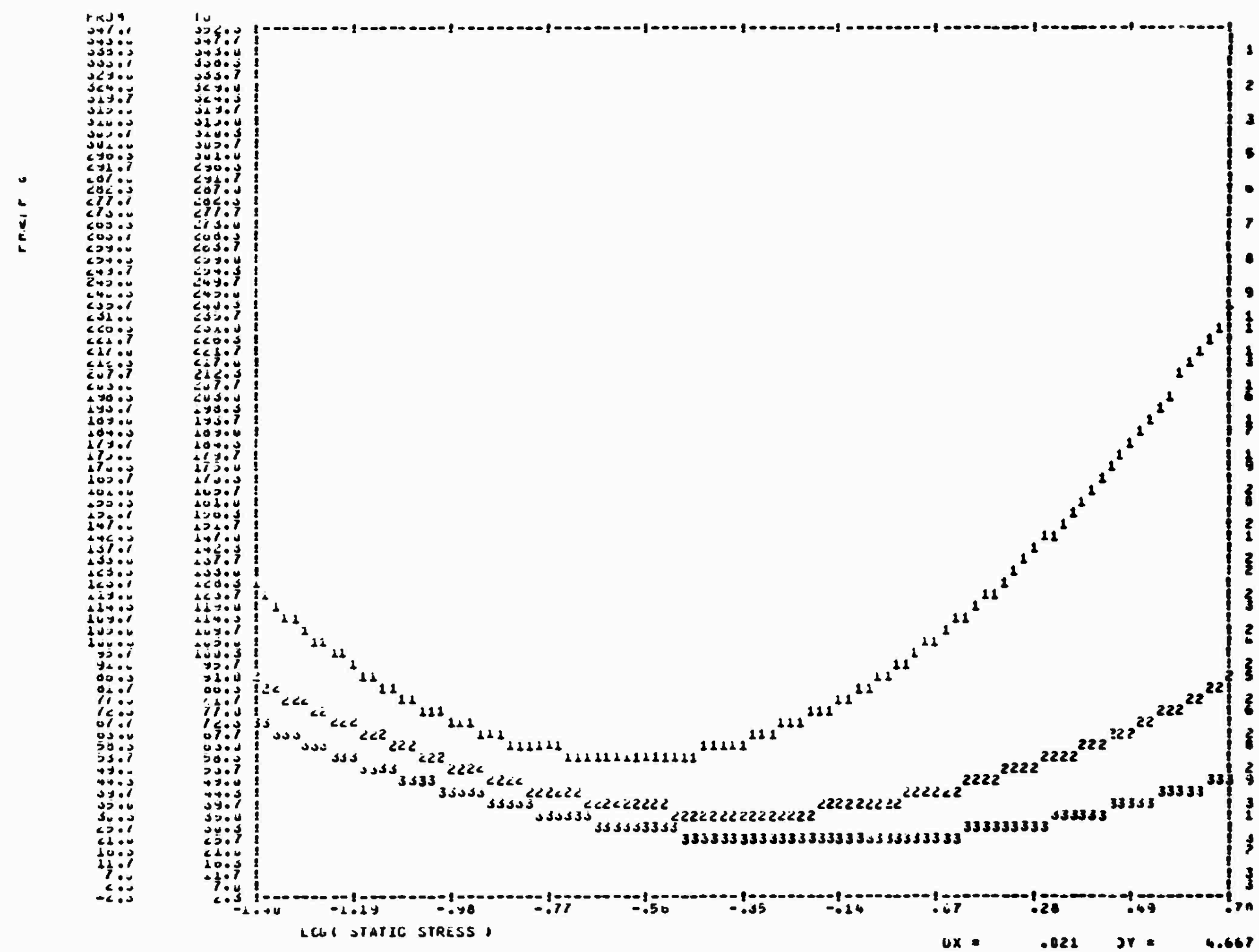


Figure D-5. 160°F, 18-inch drop height.

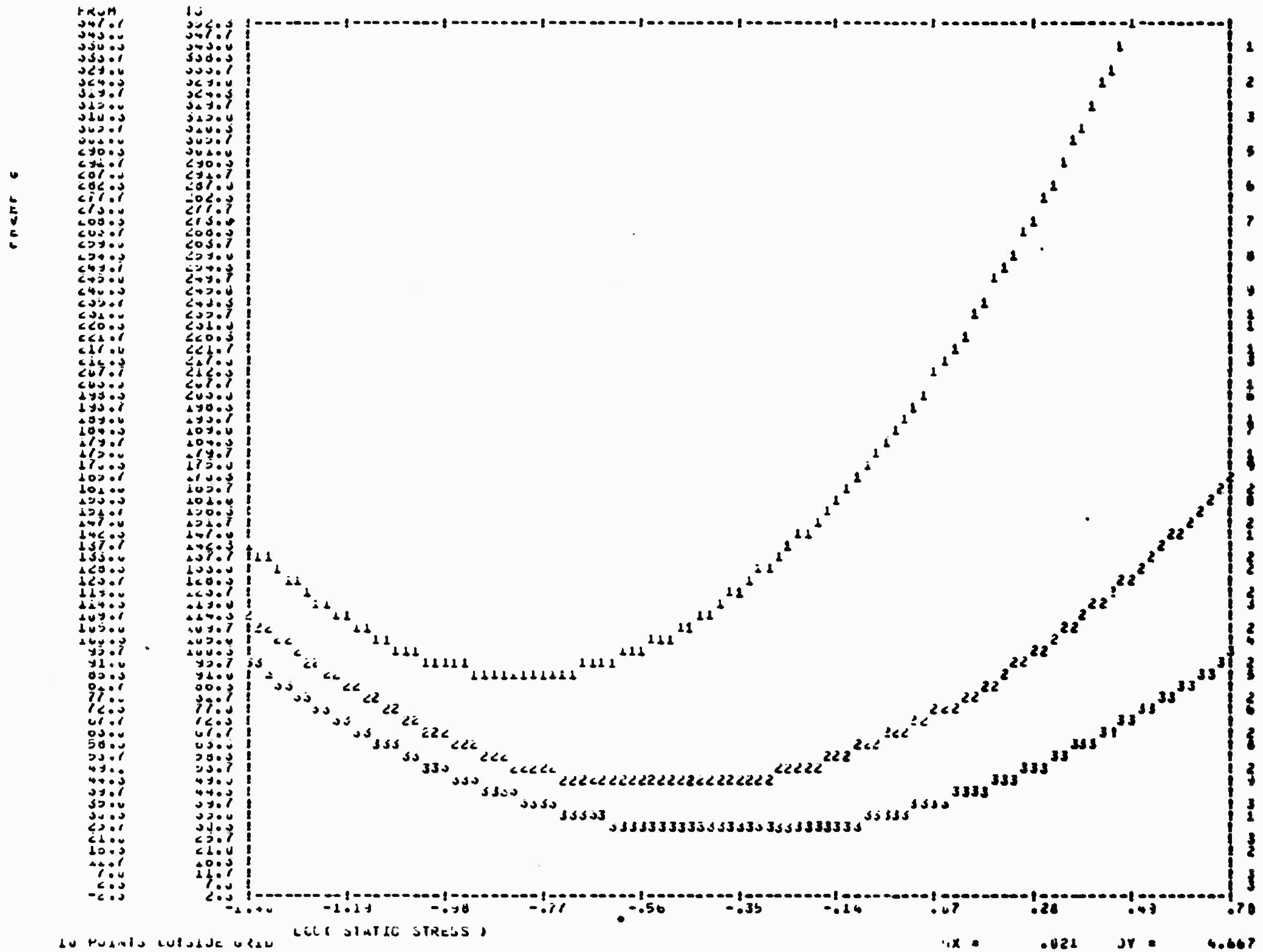


Figure D-6. 160°F, 30-inch drop height.

Appendix E

MLRD PLOTS OF DYNAMIC CUSHIONING CURVES OF THE MINICEL MODEL

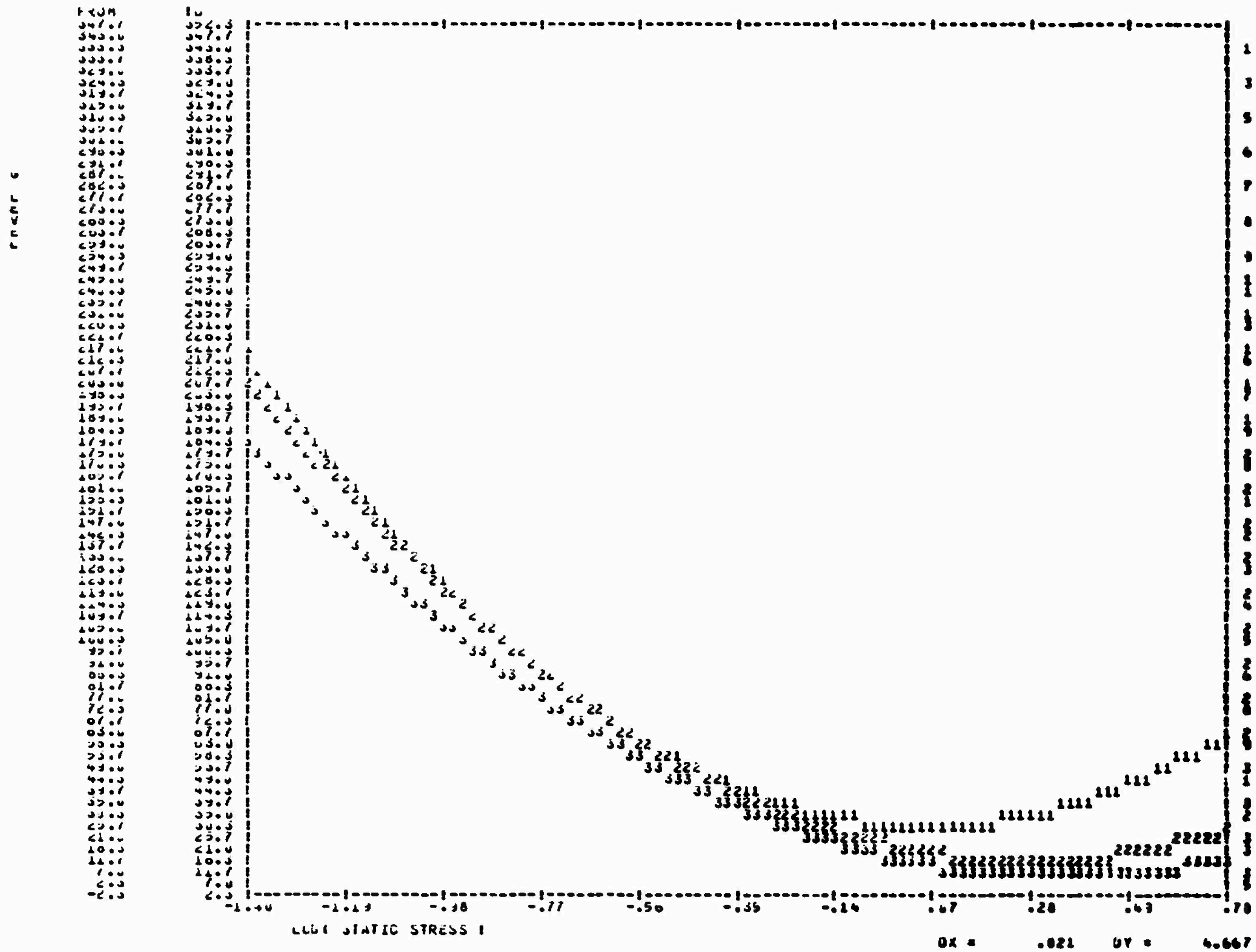


Figure E-1. -65°F, 12-inch drop height.

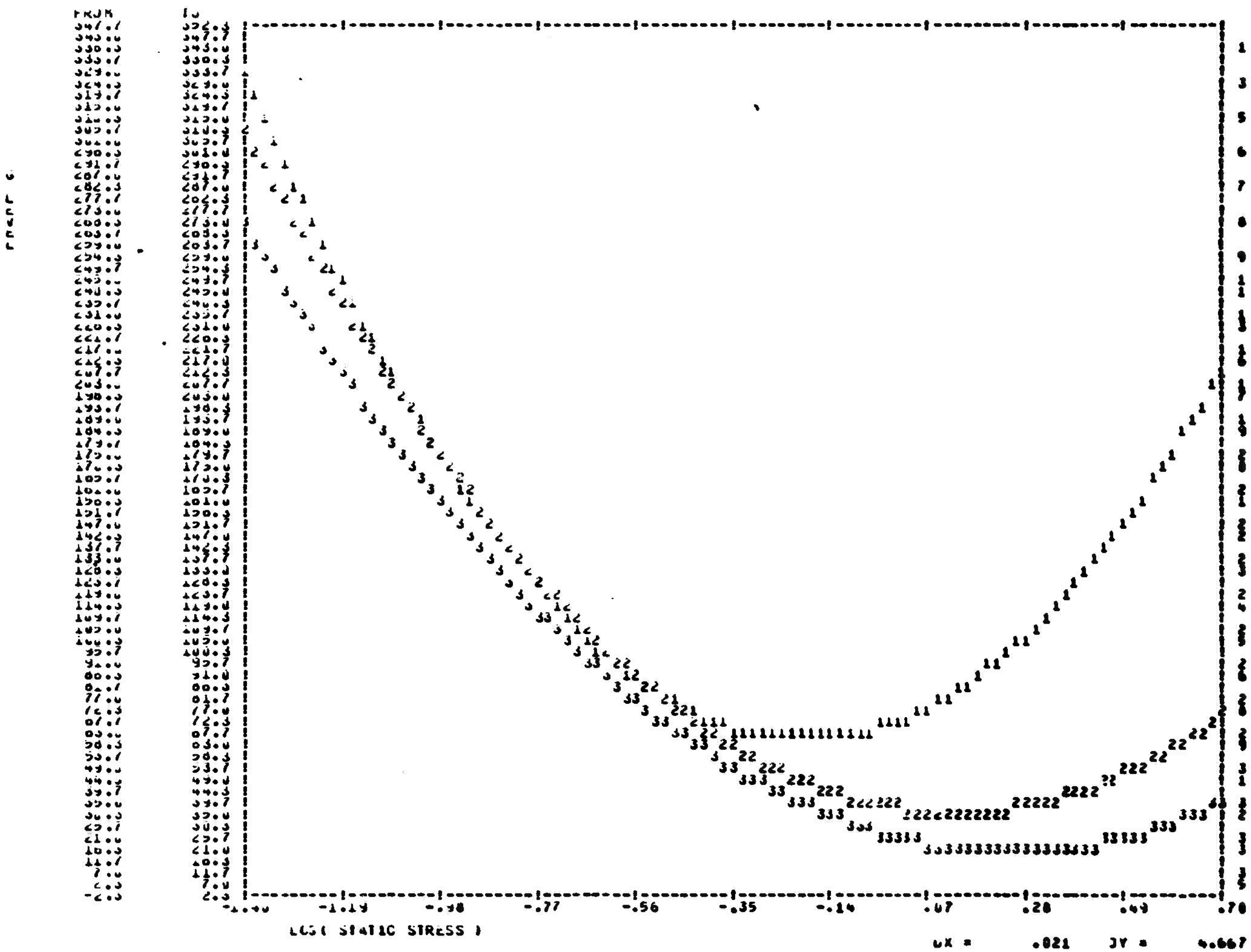


Figure E-2. -65°F, 24-inch drop height.

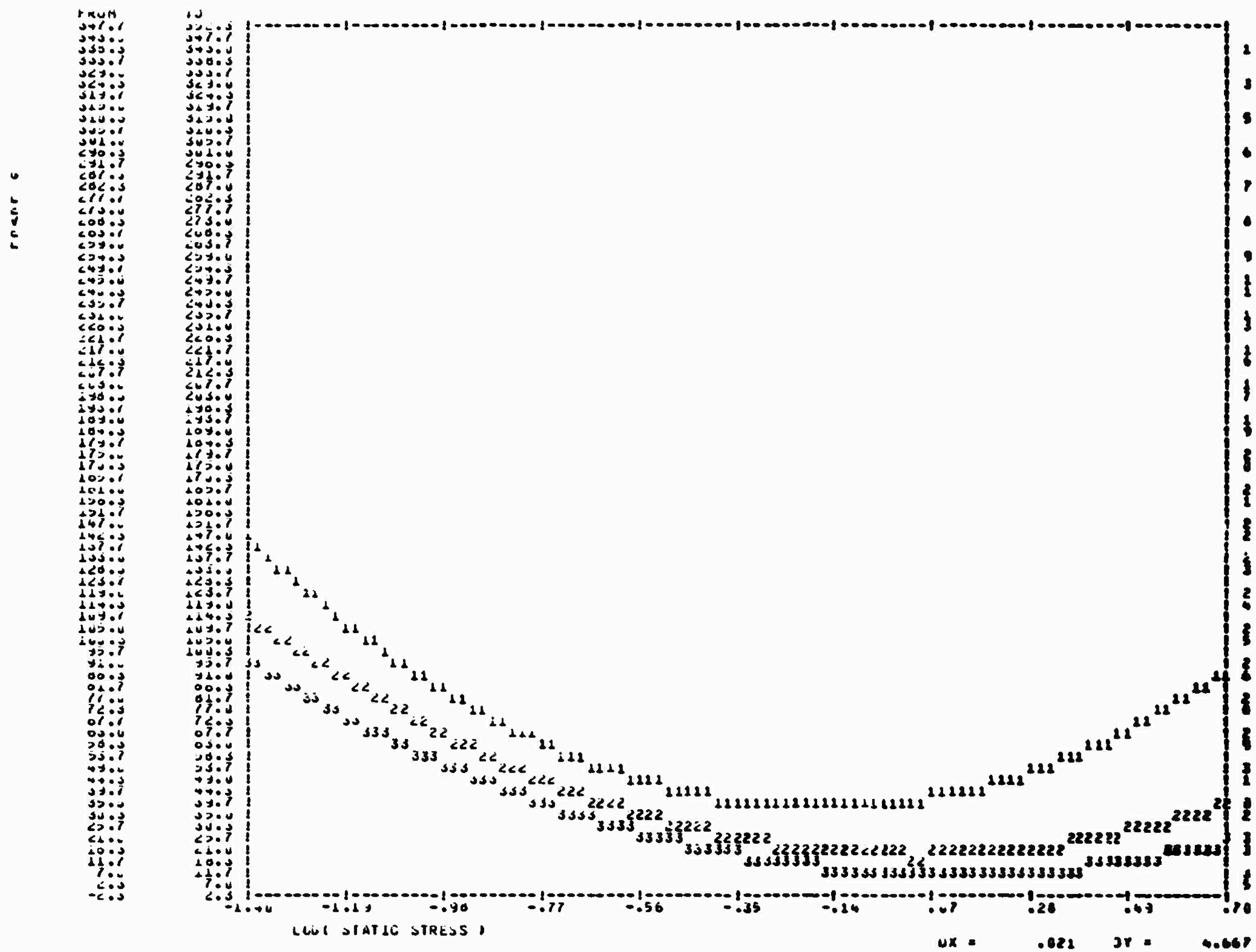


Figure E-3. 70°F, 12-inch drop height.

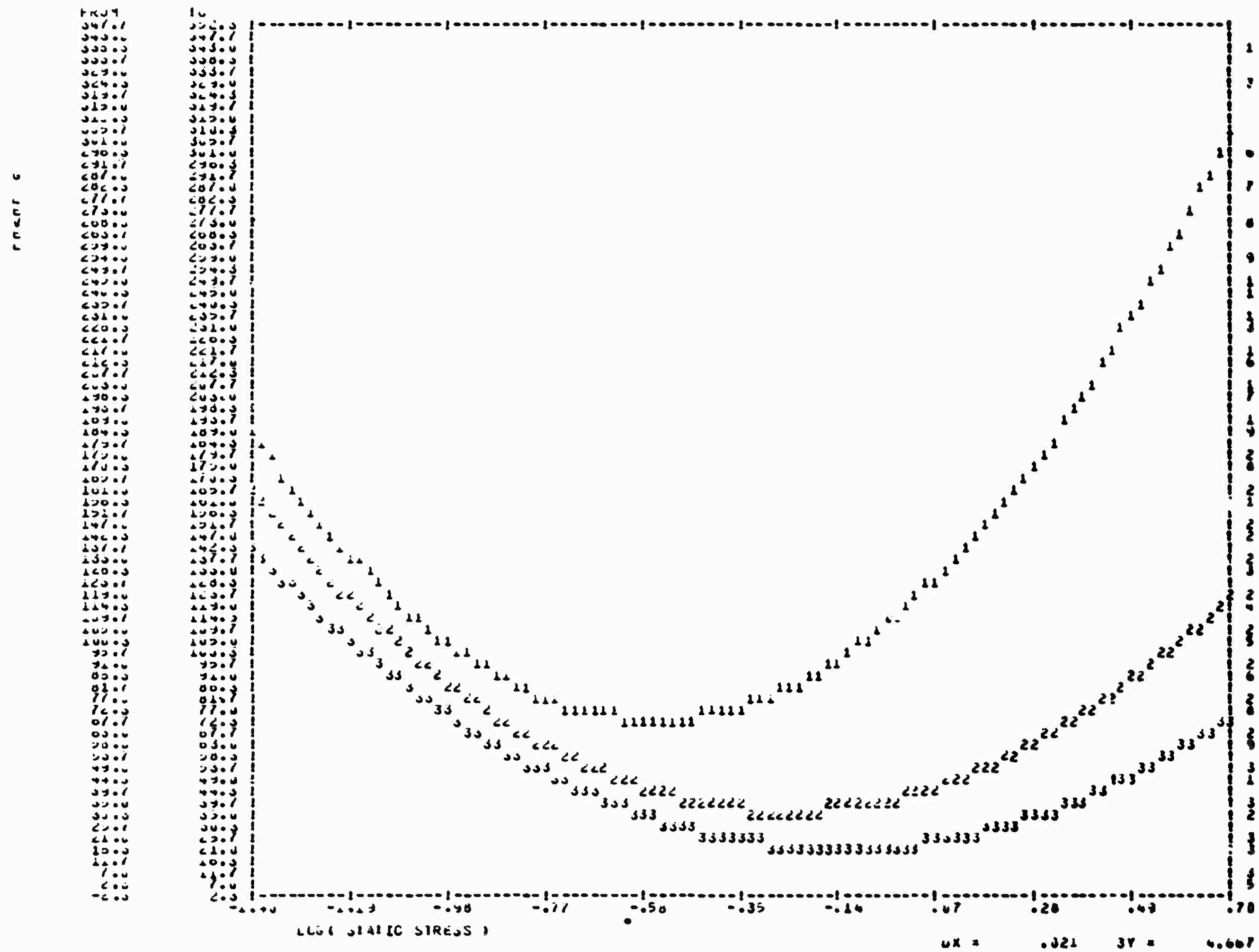


Figure E-4. 70°F, 24-inch drop height.

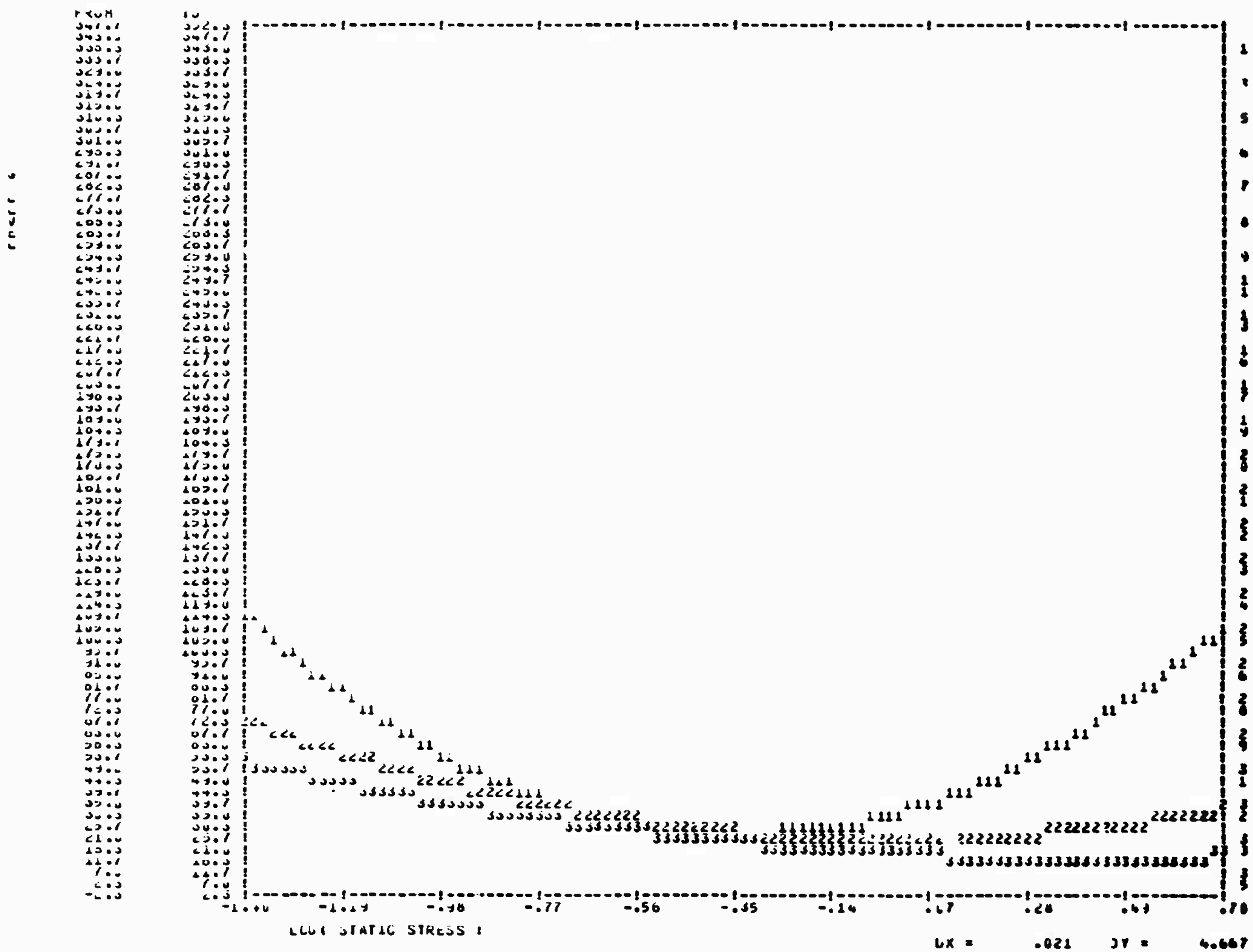


Figure E-5. 160°F, 12-inch drop height.

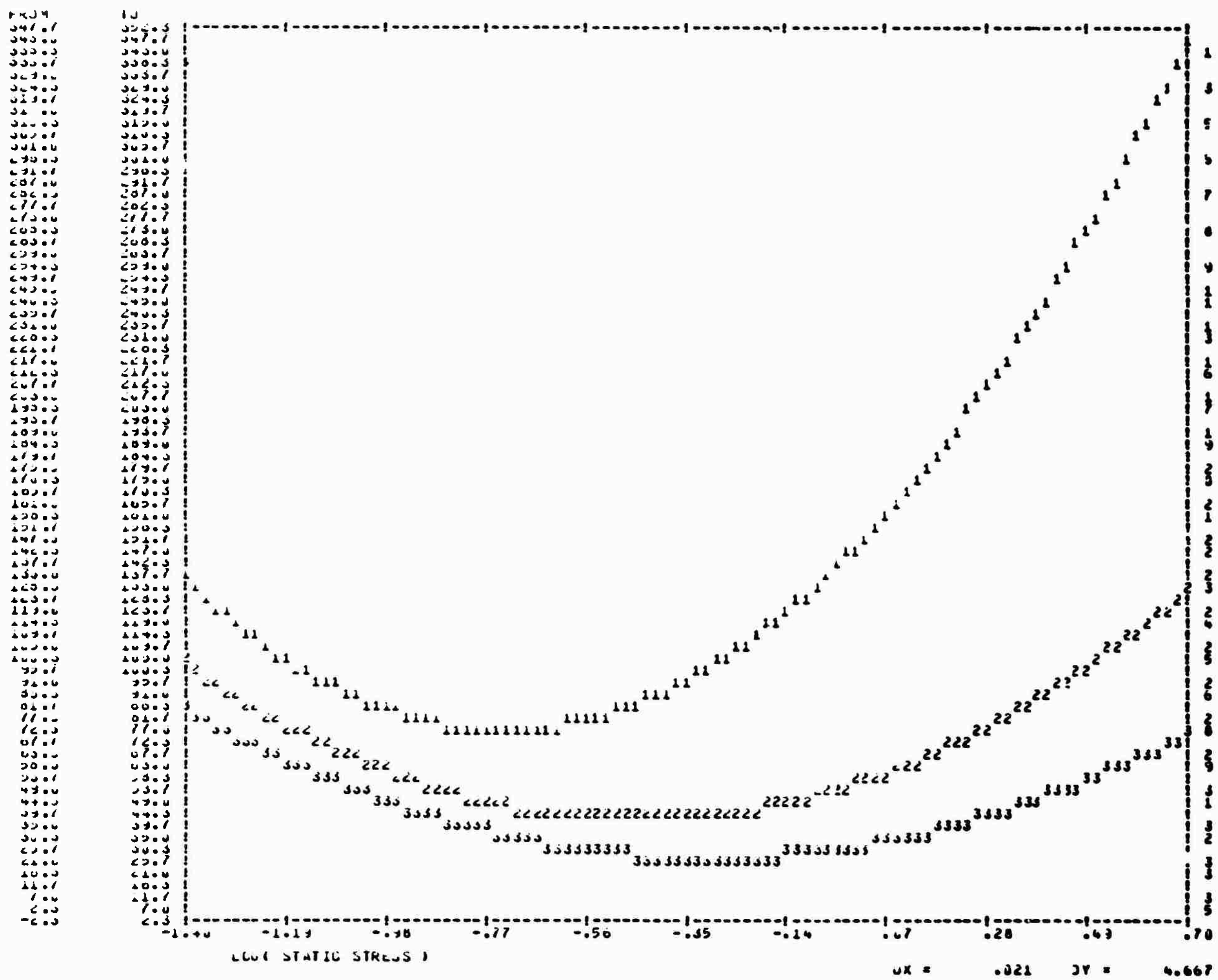


Figure E-6. 160°F, 24-inch drop height.

Appendix F

TWENTY-SEVEN-INCH DROP HEIGHT MINICEL DATA, 1, 2, AND 3 INCH THICK

Thickness	Replication	0.04			0.08			0.10			0.20			0.80		
		Temperature (°F)														
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	349	236	142	276	146	123	215	146	76	111	79	86	41	83	141
	2	301	237	123	286	148	134	203	153	88	121	67	88	40	94	148
	3	383	247	105	292	142	142	207	120	85	96	68	79	40	90	149
2"	1	321	205	110	211	106	80	189	99	73	89	55	68	31	59	56
	2	346	180	92	228	124	73	184	111	97	94	55	60	31	40	56
	3	370	191	120	205	93	94	138	98	88	98	55	61	35	49	52
3"	1	278	181	86	289	66	47	185	55	45	67	49	29	35	30	28
	2	269	171	78	293	70	52	172	60	40	61	55	31	36	30	25
	3	264	215	86	271	68	50	177	56	53	53	53	29	36	32	23

Thickness	Replication	1.00			1.50			1.50			2.60			3.00			3.40		
		Temperature (°F)																	
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160
1"	1	57	96	183	80	162	178	96	183	269	122	224	264	178	285	323	170	259	324
	2	53	92	193	73	170	108	91	173	253	131	213	292	176	251	313	158	294	337
	3	38	99	189	85	168	184	106	172	258	132	211	296	170	278	308	167	297	284
2"	1	29	29	41	30	52	73	24	72	75	37	77	94	32	85	95	42	122	97
	2	29	29	38	24	57	45	20	74	75	39	87	93	37	92	91	30	123	118
	3	17	35	47	24	56	76	25	79	77	40	76	90	40	97	91	38	120	119
3"	1	22	24	35	18	32	45	18	20	31	24	31	37	20	33	44	16	36	45
	2	24	34	33	16	21	48	18	22	29	24	29	35	17	30	62	26	34	49
	3	23	20	26	17	30	48	18	22	27	25	26	38	18	31	62	19	31	45

Appendix G

FOUR- AND FIVE-INCH THICK MINICEL DATA

MINICEL - 12 in. Drop Height

STRESS LEVELS (PSI)

Replication	0.04			0.10			0.20			0.40			0.80			
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	210	75	58	117	47	31	71	29	20	40	21	14	20	13	11	4"
2	176	72	53	118	43	30	57	28	19	38	20	16	22	12	11	
3	151	91	60	121	56	35	69	26	21	40	18	15	21	14	11	
1	155	73	48	102	41	25	53	27	16	36	15	12	19	12	9	5"
2	120	106	60	98	34	30	60	26	15	35	16	13	19	8	10	
3	135	84	41	99	39	23	54	24	18	34	41	12	21	12	10	

Replication	1.00			1.60			2.00			2.40			3.00			
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	16	13	10	13	10	15	10	11	14	10	10	13	8	10	12	4"
2	20	10	12	13	9	15	11	11	15	9	11	13	8	11	10	
3	19	13	12	12	11	12	11	11	11	11	13	12	8	9	12	
1	27	11	6	12	9	13	10	7	12	10	9	9	8	8	9	5"
2	17	10	10	4	9	11	11	9	10	8	8	10	6	8	10	
3	18	11	7	11	9	9	10	7	9	7	8	8	8	8	8	

MINICEL - 12 in. Drop Height (Continued)

STRESS LEVELS (PSI)

Replication	3.60			4.00			4.60			5.00			Thickness
	Temperature (°F)												
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	9	10	13	5	12	13	9	12	13	6	13	12	4"
2	9	10	15	9	11	13	5	12	13	5	13	21	
3	8	12	12	8	13	9	6	13	14	7	13	21	
1	12	9	15	8	8	13	6	13	10	7	8	13	5"
2	8	10	12	6	19	13	6	7	11	5	8	11	
3	7	9	10	7	9	13	7	8	13	8	10	13	

MINICEL - 18 in. Drop Height

STRESS LEVELS (PSI)

Replication	0.04			0.10			0.20			0.40			0.80			Thickness
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	235	120	61	148	58	41	80	31	26	39	22	21	24	17	19	4"
2	253	103	66	126	55	42	74	28	24	41	23	19	23	18	17	
3	206	91	66	139	55	35	71	30	25	37	23	19	25	20	16	
1	186	94	55	118	50	35	61	26	21	39	18	18	20	14	9	5"
2	198	81	62	109	46	38	59	29	21	34	22	17	23	16	15	
3	178	92	55	115	47	35	72	27	22	37	19	18	23	15	14	

MINICEL - 18 in. Drop Height (Continued)

STRESS LEVELS (PSI)

Replication	1.00			1.60			2.00			2.40			3.00			Thickness
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	21	14	16	19	15	18	12	13	16	13	16	20	11	17	19	4"
2	21	15	16	20	13	17	17	13	17	13	13	21	13	14	19	
3	22	16	18	18	12	19	14	15	18	13	18	20	11	14	19	
1	20	14	13	17	10	11	13	12	12	12	12	13	11	12	14	5"
2	19	13	13	15	12	11	13	12	14	11	12	16	12	11	14	
3	20	13	13	13	10	14	10	12	13	9	14	14	11	11	16	

Replication	3.60			4.00			4.60			5.00			Thickness
	Temperature (°F)												
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	12	18	25	11	16	25	12	24	30	13	22	28	4"
2	11	18	22	11	18	25	11	19	28	13	17	30	
3	11	15	23	10	20	25	10	19	30	11	21	31	
1	10	14	16	8	15	17	9	15	20	9	12	21	5"
2	8	11	18	9	13	19	9	13	21	12	18	20	
3	9	13	18	9	13	17	7	17	24	9	13	22	

MINICEL - 24 in. Drop Height

STRESS LEVELS (PSI)

Replication	0.04			0.10			0.20			0.40			0.80			Thickness
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	190	122	59	151	54	39	71	32	27	44	26	26	27	20	21	4"
2	215	102	61	128	53	40	73	36	29	42	25	23	26	20	22	
3	167	120	57	131	56	42	86	34	30	40	24	23	26	20	25	
1	158	102	65	141	49	39	63	31	24	48	22	22	22	18	18	5"
2	187	93	59	174	55	38	64	30	25	35	21	21	29	18	17	
3	237	90	63	102	53	35	71	32	24	42	24	19	28	16	18	

Replication	1.00			1.60			2.00			2.40			3.00			Thickness
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	
1	18	21	21	18	19	25	17	20	26	19	23	28	22	34	40	4"
2	20	19	24	18	20	23	16	20	26	18	26	26	16	28	30	
3	23	22	23	16	22	24	16	26	25	17	24	32	18	27	34	
1	21	15	18	15	15	19	14	18	21	12	16	21	12	18	24	5"
2	23	15	20	14	15	19	14	14	20	13	19	22	14	18	21	
3	21	17	17	15	14	19	13	18	21	14	16	22	12	20	24	

MINICEL - 24 in. Drop Height (Continued)

STRESS LEVELS (PSI)

Replication	3.60			4.00			
	Temperature (°F)						
	-65	70	160	-65	70	160	Thickness
1	20	32	38	12	31	56	4"
2	20	32	43	21	36	40	
3	17	33	36	22	32	42	
1	14	20	24	12	19	29	5"
2	17	18	30	15	23	25	
3	12	21	28	30	25	30	

MINICEL - 30 in. Drop Height

STRESS LEVELS (PSI)

Replication	0.04			0.10			0.20			0.40			0.80			
	Temperature (°F)															
	-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
1	274	122	75	132	56	47	69	36	33	42	27	25	26	23	27	4"
2	250	126	74	151	60	48	72	37	35	43	29	27	26	27	28	
3	233	106	83	130	57	45	63	37	32	44	28	26	26	24	29	
1	60	96	69	126	56	39	81	33	27	43	24	20	26	18	22	5"
2	252	123	68	136	54	41	63	35	27	40	25	23	25	18	21	
3	222	100	74	152	62	36	87	40	27	49	23	21	27	19	21	

MINICEL - 30 in. Drop Height (Continued)

STRESS LEVELS (PSI)

		1.00			1.60			2.00			2.40			3.00			
		Temperature (°F)															
		-65	70	160	-65	70	160	-65	70	160	-65	70	160	-65	70	160	Thickness
Replication	1	25	27	29	18	27	35	15	29	35	23	36	44	24	39	42	4"
	2	23	36	30	21	38	36	15	32	38	24	33	43	24	43	43	
	3	23	24	30	21	31	33	18	30	32	19	32	41	25	39	42	
	1	30	18	25	18	16	22	22	18	20	12	19	23	15	22	30	5"
	2	26	21	21	17	19	26	15	18	24	12	15	25	18	24	28	
	3	23	13	23	19	18	20	12	25	22	16	20	26	16	20	32	

		3.60			4.00			
		Temperature (°F)						
		-65	70	160	-65	70	160	Thickness
Replication	1	23	40	55	29	51	61	4"
	2	24	42	56	24	52	57	
	3	41	42	58	32	47	57	
	1	16	27	38	15	27	40	5"
	2	15	28	39	15	32	46	
	3	14	29	40	14	52	36	

Appendix H

PROGRAM LISTING OF CUSHION OPT

```
      PROGRAM SEARCH( INPUT=65, OUTPUT,  TAPE5=INPUT,TAPE6=OUTPUT )
      COMMON  TP, OH, TC, SS, GL, NVR, V(51), INC(51),COEFF(51),CONST,NV
      COMMON  CH, CA, CC, DROPH, GLMAX, SSL(3), SSU(3), TYPEM(2)
      DIMENSION  TT(5), PRV(5), NCK(5), TCC(3)
      EQUIVALENCE (TT(1),CH)
CCC   EQUIVALENCE (TT(1),CH),(TT(2),CA),(TT(3),CC),(TT(4),DROPH)
CCC   EQUIVALENCE (TT(5),GLMAX)

      DATA  PRV/160.,70.,-65., 30.,75./
      CALL  ID4020( 14, 14HWAYNE L. JONES )
      CALL DATE( DAY )
      DSS = 0.05
      SSMIN = 0.05
      SSMAX = 5.00
      DTC = 0.5
      TCMIN = 1.0
      TCMAX = 6.0
C
      READ 901, TYPEM
  901 FORMAT(8A10)
      PRINT 900, DAY,  TYPEM
  900 FORMAT(*1MCOANIEL SEARCH PROGRAM*20X,A10 //1X,2A10.)
C           READ EQUATION COEFFICIENTS
      READ 908, J, CONST, NV
  908 FORMAT( I5, E20.8, 5X, I5 )
      PRINT 910, CONST
  910 FORMAT(9X*INDEX  COEFFICIENT*/10X*0*F15.8)
      DO 20 J=1,NV
      READ 908, IND(J), COEFF(J)
   20 PRINT 912, J, IND(J), COEFF(J)
  912 FORMAT(1X(5,I5,E15.8)
C
   10 READ 902, DROPH,GLMAX, CH,CA,CC,     (NCK(J),J=4,5),(NCK(J),J=1,3)
  902 FORMAT(5F10.0, T1,5A10)
      IF( CCF( 5      )  .NE.  0.0 )  GO TO 800
C
      DO 15 J=1,5
      IF( NCK(J) .EQ. 1H  )  TT(J) = PRV(J)
   15 PRV(J) = TT(J)
      DH = DROPH
      PRINT 900, DAY,  TYPEM
      PRINT 904,  DH, GLMAX
  904 FORMAT(* DROP HEIGHT =*F4.0,5X*GLMAX =*F5.0/)
      PRINT 906, ( TT(J), J=1,3 )
  906 FORMAT(* TEMPERATURES =* 3F5.0/)
C
C           INITIALIZE THICKNESS
      TC = TCMIN - DTC
C           HAS MAXIMUM THICKNESS BEEN REACHED ?
  100 IF( TC .GE. TCMAX )  GO TO 500
C          INCREMENT THICKNESS
      TC = TC + DTC
      DO 120 J=1,3
  120 SSL(J) = SSU(J) = TCC(J) = 0.0
C
C           INITIALIZE TEMPERATURE INDEX
      N = 0
```

```
PROGRAM SEARCH      74/74    OPT=1                      FTN 4.2+74278      11/25/74  15.42.24.

      C          HAS THIRO TEMPERATURE BEEN REACHEO ?
        200 IF( N .GE. 3 )  GO TO 500
      C        INCFEMENT TEMPERATURE
            N = N + 1
            TP = TT(N)
      C
      C          INITIALIZE STATIC STRESS
            SS = SSMIN - OSS
      C          HAS MAXIMUM STATIC STRESS BEEN REACHEC ?
        300 IF( SS .GE. SSMAX )  GC TO 100
      C        INCPEMENT STATIC STFESS
            SS = SS + OSS
      C
            CALL MODEL
            IF( GL .GE. GLMAX )  GO TO 300
            SSL(N) = SS
            TCC(N) = TC
      C
      C          HAS MAXIMUM STATIC STRESS BEEN REACHEC ?
        400 IF( SS .GE. SSMAX )  GO TO 420
      C        INCPEMENT STATIC STRESS
            SS = SS + OSS
      C
            CALL MODEL
            IF( GL .LE. GLMAX )  GO TO 400
        420 SSU(N) = SS
      C          MAKE RANGE TEST
            IF( SSU(N)-SSL(N) .GE. 0.2 )  GO TO 200
            GO TO 100
      C
        500 CONTINUE
            PRINT 920, ( TT(J), SSL(J), SSU(J), TCC(J), J=1,3 )
        920 FORMAT(//* TEMP   SSL   SSU   TC*//(1XF4.0,F6.2,F6.2,F5.1))
            IF( SSL(3)  .EQ. 0.0 )  GO TO 510
            TEST = SSU(1) - SSL(3)
            IF( TEST .GE. 0.2 )  GO TO 520
        510 PRINT 922
        922 FORMAT(//* RANGE TEST FAILEO*)
            IF( TC .GF. TCMAX )  PRINT 924
        924 FORMAT(//* USE OTHER MATERIAL*)
            IF( TC .GE. TCMAX )   GO TO 10
            GO TO 100
        520 CONTINUE
            PRINT 926, TC, TYPEM, GLMAX, SSL(3), SSU(1)
        926 FORMAT(/1XF3.1,* INCHES OF *2410/  * GIVES A*F5.0* G PROTECTION*/
           * * USING A STATIC STRESS*/* RANGE OF*F5.2* TO*F5.2)
      C          FLCT CURVES THAT PASS RANGE TEST
            CALL MPLOT
      C
            GO TO 10
        800 CONTINUE
            CALL SMXYV( 0 ,0 )
            CALL USER ID ( 25HQUICKPL TEKTRONICS TEST   )
            CALL TEKTEST
            CALL END JOB
            END
```

```
      SUBROUTINE MODEL
      COMMON  TP, DH, TC, SS, GL, NVR, V(51), IND(51),COEFF(51),CONST,NV
      COMMON  CH, CA, CC, DRCPH, GLMAX, SSL(3), SSU(3), TYPE*(2)
C
C********* DYNAMIC CUSHIONING MODEL ******************************************
C
      SS100 = SS * 100.
      AL = ALOG( SS100 )
      AL2 = AL * AL
      SRDH = SQRT( DH )
      TCSH = TC ** (-3.5)
       TR = (TP+460)/100.
      TR2 = TR * TR
      TR3 = TR * TR2
      TR4 = TR3 * TR
C
      TCOH = TC ** (-0.5)
      TCTH = TC ** (-1.5)
      TCINV= TC ** (-2.5)
C
      V(01) = TR    * TCOH    * 1.0
      V(02) = TR    * TCOH    * 1.0      * AL
      V(03) = TR    * TCOH    * 1.0      * AL2
      V(04) = TR    * TCTH    * SRDH
      V(05) = TR    * TCTH    * SRDH     * AL
      V(06) = TR    * TCTH    * SRDH     * AL2
      V(07) = TR    * TCOH    * SRDH
      V(08) = TR    * TCOH    * SRDH     * AL
      V(09) = TR    * TCOH    * SRDH     * AL2

      V(10) = TR2   * TCOH    * 1.0
      V(11) = TR2   * TCOH    * 1.0      * AL
      V(12) = TR2   * TCOH    * 1.0      * AL2
      V(13) = TR2   * TCTH    * SRDH
      V(14) = TR2   * TCTH    * SRDH     * AL
      V(15) = TR2   * TCTH    * SRDH     * AL2
      V(16) = TR2   * TCOH    * SRDH
      V(17) = TR2   * TCOH    * SRDH     * AL
      V(18) = TR2   * TCTH    * SRDH     * AL2

      V(19) = TR3   * TCOH    * 1.0
      V(20) = TR3   * TCOH    * 1.0      * AL
      V(21) = TR3   * TCOH    * 1.0      * AL2
      V(22) = TR3   * TCTH    * SRDH
      V(23) = TR3   * TCTH    * SRDH     * AL
      V(24) = TR3   * TCTH    * SRDH     * AL2
      V(25) = TR3   * TCOH    * SRDH
      V(26) = TR3   * TCOH    * SRDH     * AL
      V(27) = TR3   * TCTH    * SRDH     * AL2

      V(28) = TR    * TCTH
      V(29) = TR    * TCTH               * AL
      V(30) = TR    * TCTH               * AL2
      V(31) = TR2   * TCTH
      V(32) = TR2   * TCTH               * AL
      V(33) = TR2   * TCTH               * AL2
      V(34) = TR3   * TCTH
```

```
SUBROUTINE MODEL       74/74    OPT=1                      FTN 4.2+74278      11/25/74  15.42.21.

      V(35) = TR3    * TCTH                * AL
      V(36) = TR3    * TCTH                * AL2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C         COMPUTE DYNAMIC CUSHIONING FUNCTION
      GL = CONST
      DO 100 J=1,NV
      I = INC(J)
  100 GL = GL + COEFF(J) * V(I)
      RETURN
      END
```

```
SUBROUTINE MPLOT       74/74    OPT=1                       FTN 4.2+74278      11/25/74  15.42.41.

                    SUBROUTINE MPLOT
                    COMMON  TP, CH, TC, SS, GL, NVR, V(51), IND(51),COEFF(51),CCAST,NV
                    COMMON  CH, CA, CC, DROPH, GLMAX, SSL(3), SSU(3), TYPEM(2)
                    DIMENSION        X(101), Y(101,3), A(855), GG(101) , S(101)
                    DIMENSION  HEAD(10), LEFT(5), BOTTOM(8), NCH(3),  T(3), LAB(3,8)
                    DATA  NCH/1HH,1HA,1HC/
                    DATA  LEFT /2*1H , 7HG LEVEL, 2*1H  /
                    DATA  BOTTOM /2*1H ,10HSTATIC STR ,3HESS, 4*1H  /
                    DATA  HEAD(1)/25H DYNAMIC CUSHIONING CURVE /
              C
                    T(1) = CH
                    T(2) = CA
                    T(3) = CC
                    HEAD(4) = TYPEM(1)
                    HEAD(5) = TYPEM(2)
                    XMIN = ALOG10( 0.04 )
                    XMAX = ALOG10( 5.00 )
                    DX = ( XMAX - XMIN ) / 100.
              C
                    XX= XMIN
                    DO 10 JP = 1, 101
                    X(JP)= XX
                    S(JP) = 10.  ** XX
                    XX = XX + DX
                    GG(JP) = GLMAX
                 10 CONTINUE
              C
                    DO 30 K=1,3
                    TP = T(K)
                    DO 20 JP = 1,101
                    SS = S(JP)
                    CALL MODEL
                    Y(JP,K) = GL
                 20 CONTINUE
                 30 CONTINUE
              C
                    CALL SETGRID( A, -76, XMIN, XMAX, 0.0, 350. )
                    CALL LABGRID( A, 1, 25, 25H      LOG( STATIC STRESS )  )
                    CALL LABGRID( A, 2, 30, LEFT )
                    CALL LABGRID( A, 3,  40, HEAD )
              C
                    CALL PLTGRID( A, 1H., 101, X, GG       )
                    CALL PLTGRID( A, 1HH, 101, X, Y(1,1)   )
                    CALL PLTGRID( A, 1HA, 101, X, Y(1,2)   )
                    CALL PLTGRID( A, 1HC, 101, X, Y(1,3)   )
              C
                    ENCODE( 120,904,LAB )  CH, CA, CC, DROPH
                904 FORMAT(*H =*F5.0* DEGREES*14X, *A =*F5.0* DEGREES*14X, *C =*F5.0,
                   1* DEGREES*14X,*DROP HEIGHT =*F4.0* INCHES*6X)
                    ENCODE( 120,906,LAB(1,5) ) TC,TYPEM, GLMAX, SSL(3), SSL(1)
                906 FORMAT(F3.1* INCHES OF *A10,A6,*GIVES A*F5.0* G PROTECTION*5X,
                   1 *USING A STATIC STRESS*9X,*RANGE OF*F5.2* TO*F5.2,9X)
                    NY = 76
                    DO 40 J=1,8
                    NY = NY - 1
                    CALL REMGRID( A, 40, NY, 25, LAB(1,J) )
                 40 CONTINUE
```

```
SUPPOUTINE MPLOT       74/74    OPT=1                         FTN 4.2+74278      11/25/74  15.42.41.

C
      CALL PRINTPL( A, 6LOUTPUT )
C
      CALL SC4020
      CALL LABEL( 1, 35, BOTTOM, 5 )
      CALL LABEL( 1, 30, LEFT  , 6 )
      CALL LABEL( 1,  50, HEAD , 3 )
      CALL PLOTA( S, GG, 0.04,5.0 , 0.,350., 101, 5, 4, 1, 100 )
      JX = 420
      JY = 900.
      DO 45 J=1,8
      CALL RNOTE( JX, JY, 30, LAB(1,J) )
   45 JY = JY - 25
      DO 50 K = 1,3
      CALL PLOT3( S, Y(1,K), 101, -0 )
      CALL NOTE( S(100), Y(098,K), 1, NCH(K) )
   50 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TEKTEST
      DIMENSION  X(10),Y(10),Z(10)
      DATA  X/1., 2., 3., 4., 5., 6., 7., 8., 9., 10. /
      DATA  Y/100.,50.,70.,50.,60.,50.,70.,50.,70., 40./
      DATA  Z/100.,100.,100.,80.,90.,80.,90.,80.,70.,100./
      CALL QUICKPL(X,Y,10,-2,6HX-AXIS,6HY-AXIS,8H TEKTEST,0.,0.,X(10),
     * Y(10),Y(1),200,1)
      CALL QUICKPL(X,Z,-10,-1)
      CALL PRINTV( 4,4HXXXX,500,500)
      CALL PRINTV( 4,4H0000,600,600)
      CALL LINEV( 0,500,500,500)
      CALL LINEV(500,0 ,500,500)
      CALL LINEV( 0,600,600,600)
      CALL LINEV(600,0 ,600,600)
      RETURN
      END
```